



PATENT

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

Applicants: Min-Goo KIM Examiner: VLAHOS, Sophia
Serial No: 09/986,210 Group Art Unit: 2611
Filed: October 22, 2001 Docket: 678-762 (P9997)
Dated: September 26, 2007

For: **METHOD AND DEVICE FOR TRANSMITTING PACKET DATA IN MOBILE COMMUNICATION SYSTEM**

Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

TRANSMITTAL OF PRIORITY DOCUMENT

Sir:

Enclosed is a certified copy of Korean Appln. No. 10-2000-0062153 filed on October 21, 2000, from which priority is claimed under 35 U.S.C. §119.

Respectfully submitted,

Paul J. Farrell
Registration No. 33,494
Attorney for Applicants

THE FARRELL LAW FIRM
333 Earle Ovington Boulevard, Suite 701
Uniondale, New York 11553
(516) 228-3565

CERTIFICATE OF MAILING UNDER 37 C.F.R. § 1.8 (a)

I hereby certify that this correspondence is being deposited with the United States Postal Service as first class mail, postpaid in an envelope, addressed to the: Commissioner of Patents, P.O. Box 1450, Alexandria, VA 22313-1450 on September 26, 2007.

Dated: September 26, 2007

Patrick Y. Mirville



별첨 사본은 아래 출원의 원본과 동일함을 증명함.

This is to certify that the following application annexed hereto is a true copy from the records of the Korean Intellectual Property Office.

출원 번호 : 10-2000-0062153

Application Number

출원 년 월 일 : 2000년 10월 21일

Filing Date OCT 21, 2000

출원인 : 삼성전자주식회사

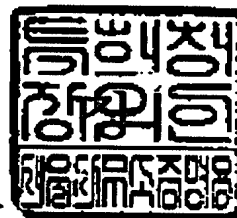
Applicant(s) SAMSUNG ELECTRONICS CO., LTD.



2007년 08월 20일

특허청

COMMISSIONER



◆ This certificate was issued by Korean Intellectual Property Office. Please confirm any forgery or alteration of the contents by an issue number or a barcode of the document below through the KIPOnet- Online Issue of the Certificates' menu of Korean Intellectual Property Office homepage (www.kipo.go.kr). But please notice that the confirmation by the issue number is available only for 90 days.

【서지사항】

| | |
|------------|---|
| 【서류명】 | 서지사항 보정서 |
| 【수신처】 | 특허청장 |
| 【제출일자】 | 2001.01.05 |
| 【제출인】 | |
| 【명칭】 | 삼성전자 주식회사 |
| 【출원인코드】 | 1-1998-104271-3 |
| 【사건과의 관계】 | 출원인 |
| 【대리인】 | |
| 【성명】 | 이건주 |
| 【대리인코드】 | 9-1998-000339-8 |
| 【포괄위임등록번호】 | 1999-006038-0 |
| 【사건의 표시】 | |
| 【출원번호】 | 10-2000-0062153 |
| 【출원일자】 | 2000.10.21 |
| 【발명의 명칭】 | 복합 재전송형식을 사용하는 데이터 통신시스템의 패킷 전송 장치 및 방법 |
| 【제출원인】 | |
| 【접수번호】 | 1-1-2000-0221412-62 |
| 【접수일자】 | 2000.10.21 |
| 【보정할 서류】 | 특허출원서 |
| 【보정할 사항】 | |
| 【보정대상항목】 | 발명자 |
| 【보정방법】 | 정정 |
| 【보정내용】 | |
| 【발명자】 | |
| 【성명의 국문표기】 | 김민구 |
| 【성명의 영문표기】 | KIM,Min Goo |

【주민등록번호】 640820-1XXXXXX

【우편번호】 442-470

【주소】 경기도 수원시 팔달구 영통동 973-3 우성아파트 822-406

【국적】 KR

【발명자】

【성명의 국문표기】 하상혁

【성명의 영문표기】 HA,Sang Hyuck

【주민등록번호】 730219-1XXXXXX

【우편번호】 442-809

【주소】 경기도 수원시 팔달구 영통동 948-4 황골마을
주공아파트108 -2002

【국적】 KR

【취지】 특허법시행규칙 제13조의 규정에 의하여 위와 같이 제출합니다.

대리인

이건주 (인)

【수수료】

【보정료】 0 원

【기타 수수료】 원

【합계】 0 원

【첨부서류】 1.기타첨부서류_1통[확인서]

【서지사항】

【서류명】 특허출원서
【권리구분】 특허
【수신처】 특허청장
【참조번호】 0007
【제출일자】 2000. 10. 21
【국제특허분류】 H04M
【발명의 국문명칭】 복합 재전송형식을 사용하는 데이터 통신시스템의 패킷 전송 장치 및 방법
【발명의 영문명칭】 Packet Transmission Apparatus and Method of Hybrid Automatic Repeat Request Data Communication Systems
【출원인】
【명칭】 삼성전자 주식회사
【출원인코드】 1-1998-104271-3
【대리인】
【성명】 이건주
【대리인코드】 9-1998-000339-8
【포괄위임등록번호】 1999-006038-0
【발명자】
【성명의 국문표기】 김민구
【성명의 영문표기】 KIM, Min-Koo
【주민등록번호】 640820-1XXXXXXX
【우편번호】 442-470
【주소】 경기도 수원시 팔달구 영통동 973-3 우성아파트 822-406
【국적】 KR
【취지】 특허법 제42조의 규정에 의하여 위와 같이 출원합니다.

대리인

이건주 (인)

【수수료】

【기본출원료】 20 면 29,000 원

【가산출원료】 28 면 28,000 원

【우선권주장료】 0 건 0 원

【심사청구료】 0 항 0 원

【합계】 57,000 원

【첨부서류】 1. 요약서 · 명세서(도면)_1통

【요약서】

【요약】

본 발명은 데이터 통신시스템의 전송 방식에 관한 것으로, 특히 복합 재전송 형식을 사용하는 통신시스템에서 전송 효율을 극대화하기 위한 패킷 전송 장치 및 방법이 개시되어 있다. 이러한 본 발명은, 복합 재전송형식을 사용하는 통신시스템에서 전송 효율을 극대화하기 위한 패킷 전송 방법에 있어서, 하나의 물리계층 패킷(PLP) 전송에 사용되는 모든 연속한 전송 프레임의 총 심볼 수를 각각의 서브 부호의 부호어 심볼 수에 최대로 할당되도록 분할하는 과정과, 각 서브 부호의 부호율에 의해 발생하는 심볼 수가 각 전송 프레임에 할당된 심볼 수보다 크도록 설정하고, 전송 프레임에 배치되지 못한 나머지 심볼을 삭제하여 전송하는 과정을 포함한다.

【대표도】

도 21

【색인어】

HARQ, TUTBO CODES, PACKET COMBINING, SOFT COMBINING, THROUGHPUT, CODE COMBINING

【명세서】**【발명의 명칭】**

복합 재전송형식을 사용하는 데이터 통신시스템의 패킷 전송 장치 및 방법
{Packet Transmission Apparatus and Method of Hybrid Automatic Repeat Request
Data Communication Systems}

【도면의 간단한 설명】

- <1> 도 1은 전형적인 IS-2000 시스템에서 사용되는 부호율 $R=1/5$ 인 터보 부호기 구성을 보여주는 도면.
- <2> 도 2는 하나의 슬롯을 통해 물리계층 패킷을 전송하는 하이브리드 ARQ 프로토콜을 보여주는 도면.
- <3> 도 3은 두개의 슬롯을 통해 물리계층 패킷을 전송하는 하이브리드 ARQ 프로토콜을 보여주는 도면.
- <4> 도 4 내지 도 20은 본 발명의 실시예에 따른 각 데이터 레이트별 전송 패킷의 구조를 보여주는 도면.
- <5> 도 21은 본 발명의 실시예에 따른 동작이 적용되는 시스템의 구성을 보여주는 도면.

【발명의 상세한 설명】

【발명의 목적】

【발명이 속하는 기술분야 및 그 분야의 종래기술】

- <6> 본 발명은 데이터 통신시스템의 전송 방식에 관한 것으로, 특히 복합 재전송 형식을 사용하는 통신시스템에서 전송 효율을 극대화하기 위한 패킷 전송 장치 및 방법에 관한 것이다.
- <7> 통상적으로 무선 통신시스템에서는 순방향 데이터 통신을 수행하는 경우, 이동국은 기지국으로부터 순방향(forward) 채널 즉, 전용채널(DCH: Dedicated Channel) 등과 같은 채널을 할당받아 데이터 패킷을 수신하게 된다. 상기 무선 통신시스템은 위성시스템, ISDN, 디지털 셀룰러(Digital cellular), W-CDMA, UMTS, IMT-2000 등을 통칭한다. 이 때, 이동국은 성공적으로 수신되어진 패킷은 상위 계층으로 전달한다. 하지만, 오류가 발생한 패킷에 대해서는 복합 재전송방식(HARQ: Hybrid Automatic Repeat Request)을 사용하여 재전송을 요청하게 된다. 상기 복합 재전송방식이란 오류 정정 부호(FEC: Forward Error Correction)와 오류검출 시에 데이터 패킷의 재전송을 요구하는 재전송방식(ARQ: Automatic Repeat Request)을 모두 사용하는 재전송방식이다. 이는 오류 검출시 채널 부호화방식(Channel coding scheme)을 사용하여 데이터 전송효율성, 즉 이득율(throughput)을 높이고 시스템의 성능을 개선하기 위한 전송방식이다. 상기 무선 통신시스템에서 주로 사용되는 채널 부호화방식으로는 컨벌루션널 부호(convolutional codes)나 터보 부호(turbo codes) 등을 FEC로 하는 채널 부호화방식이 사용되었다.

<8> 또한, 하이브리드 ARQ(Hybrid ARQ)방식을 사용하는 시스템에서 전송효율(throughput)을 개선하고자 연성 결합(Soft combining)을 사용하며 여기에는 두 가지 방식이 가능하다. 첫 번째 방식으로 packet diversity combining(패킷 다이버시티 결합)을 사용하는 구조이며, 두 번째 방식으로 packet code combining(패킷 부호 결합)을 사용하는 방식이다. 일반적으로 이 두 방식을 모두 Soft Packet Combining(연성 패킷 결합)이라고 말하며 이 중 상기 packet diversity combining 방식은 상기 packet code combining 방식에 비하여 성능면에서 sub-optimal 방식이지만 구현의 편의성 때문에 성능상의 큰 손실이 없는 경우 자주 사용되는 방식이다.

<9> 통상적으로, Packet(패킷)을 전송하는 시스템에서 전송효율(Throughput)을 증가시키기 위해서 연성 패킷 부호 결합을 사용한다. 즉, 각각의 전송마다 전달되는 Packet에 대하여 부호율이 R인 상호 다른 부호를 전송하고 복호된 결과 수신된 패킷에 오류가 검출되는 경우 이를 소멸시키지 않고 저장한 뒤에 향후 재 전송되어 올 패킷과 연성결합(Soft Combining)을 하는 방식을 말한다. 이때 재전송되는 패킷은 상호 다른 부호가 사용될 수 있다. 즉, 상기 패킷 부호 결합 방식은 부호율이 R인 N개의 패킷을 수신한 경우 각각의 패킷을 사용하여 실효 부호율(effective code rate)이 R/N 인 부호로 전환한 뒤에 복호함으로써 부호화이득(coding gain)을 얻는 효과를 갖는 방식이다.

<10> 반면에 패킷 다이버시티 결합은 각각의 전송마다 전달되는 Packet에 대하여 부호율이 R인 동일한 부호를 전송하고 복호된 결과 수신된 패킷에 오류가 검출되는

경우 이를 소멸시키지 않고 저장한 뒤에 향후 재 전송되어 올 패킷과 연성결합 (Soft Combining)을 하는 방식을 말한다. 이때 재전송되는 패킷은 항상 동일한 부호가 사용된다. 따라서 패킷 다이버시티 결합은 랜덤채널에서 일종의 Symbol Averaging (심볼에너지 평균과정)으로 볼 수 있으며 수신심볼 연성출력(soft output)을 평균함으로써 얻는 잡음전력감소 효과와 페이딩 채널에서 복수개의 심볼을 전송함으로써 다중성 채널에서 제공되는 다중성이득(diversity gain)만을 사용하는 방식이라고 볼 수 있다. 이에 반해서 패킷 부호 결합은 이러한 이득 이외에도 Code structure(코드 구조)에 따른 추가의 Coding Gain(코딩 게인)을 가지고 있다.

<11>

현재까지의 패킷통신 시스템에서는 구현의 용이함 때문에 대부분 패킷 다이버시티 결합을 사용하고 있으며 동기방식의 IS-2000 시스템과 비동기방식의 UMTS 시스템 등에서 이러한 방식이 고려되고 있다. 그러나 이는 기존의 패킷통신 시스템들이 대부분 길쌈부호를 사용하였고, 길쌈부호의 경우 부호율 R 이 낮은 부호(codes)를 사용하는 경우에 상기 패킷 다이버시티 결합을 사용한다 해도 그리 큰 이득이 제공되지 않기 때문이었다. 즉, $R=1/3$ 사용하는 시스템에서 재전송이 가능한 경우, 상기 패킷 다이버시티 결합을 사용하는 경우와 패킷 부호 결합을 사용하여 $1/6$ 부호율을 사용하는 경우의 성능차이가 그리 크지 않기 때문에 구현의 복잡도를 고려하여 패킷 다이버시티 결합을 사용하였다. 그러나 오류정정부호(Forward Error Correction Codes: FEC)로 터보 부호를 사용하는 경우에는 이러한 기존의 개념과는 다른 방식이 요구된다. 왜냐하면 터보 부호는 반복 복호(iterative decoding)에 의해서 그 성능이 Shannon limit에 근접하도록 설계된 오류정정부호이

며, 부호율에 따른 성능의 차이가 기존의 길쌈 부호(Convolutional codes)와는 달리 분명하게 존재하기 때문이다. 즉, 이러한 점을 고려할때 재전송을 사용하는 패킷통신 시스템에서는 터보 부호(Turbo codes)를 사용하는 패킷 부호 결합 방식을 구현하는 것이 성능을 향상시키는 방법이 된다.

【발명이 이루고자 하는 기술적 과제】

<12> 따라서 본 발명의 목적은 패킷 부호 결합 방식을 사용하거나 패킷 부호 결합 방식과 패킷 다이버시티 결합 방식을 함께 사용하는 시스템의 전송효율을 개선하고자 전송하는 패킷의 구조를 최적화하고 전송 물리채널의 기능블록을 최적화하는 방법 및 장치를 제공함에 있다.

<13> 본 발명의 다른 목적은 패킷 부호 결합과 패킷 다이버시티 결합을 데이터 레이트에 따라서 가변적으로 사용하는 시스템에서 각 전송 패킷의 구조를 최적화하는 방법 및 장치를 제공함에 있다.

<14> 이러한 목적들을 달성하기 위한 본 발명은, 복합 재전송형식을 사용하는 통신시스템에서 전송 효율을 극대화하기 위한 패킷 전송 방법에 있어서, 하나의 물리계층 패킷(PLP) 전송에 사용되는 모든 연속한 전송 프레임의 총 심볼 수를 각각의 서브 부호의 부호어 심볼 수에 최대로 할당되도록 분할하는 과정과, 각 서브 부호의 부호율에 의해 발생하는 심볼 수가 각 전송 프레임에 할당된 심볼 수보다 크도록 설정하고, 전송 프레임에 배치되지 못한 나머지 심볼을 삭제하여 전송하는 과정

을 포함한다.

【발명의 구성】

<15> 이하 본 발명의 바람직한 실시 예를 첨부된 도면의 참조와 함께 상세히 설명한다. 우선 각 도면의 구성요소들에 참조부호를 부가함에 있어서, 동일한 구성요소들에 한해서는 비록 다른 도면상에 표시되더라도 가능한 동일 부호를 가지도록 하였다. 또한 본 발명을 설명함에 있어서, 관련된 공지기능 혹은 구성에 대한 구체적인 설명이 본 발명의 요지를 불필요하게 흐릴 수 있다고 판단된 경우 그 상세한 설명은 생략한다.

<16> 1. Packet Code Combining

<17> 패킷을 전송하는 시스템에서는 전송효율(Throughput)을 증가시키기 위해서 소프트 패킷 부호 결합(Soft Packet Code Combining) 방식을 사용한다. 즉, 상기 소프트 패킷 부호 결합 방식은 각각의 전송마다 전달되는 패킷에 대하여 부호율이 R인 상호 다른 부호를 전송하고, 복호된 결과 수신된 패킷에 오류가 검출되는 경우 이를 소멸시키지 않고 저장한 뒤에 향후 재 전송되어 올 패킷과 연성결합(Soft Combining)을 하는 방식을 말한다. 이때 재전송되는 패킷은 상호 다른 부호가 사용될 수 있다. 즉, 부호율이 R인 N개의 패킷을 수신한 경우 각각의 패킷을 사용하여 실효 부호율(effective code rate)이 R/N 인 부호로 전환한 뒤에 복호함으로써 부호

화 이득(coding gain)을 얻는 효과를 주는 방식이 패킷 부호 결합 방식이다.

<18> 반면에, 패킷 다이버시티 결합(Packet Diversity Combining) 방식은 각각의 전송마다 전달되는 패킷에 대하여 부호율이 R 인 동일한 부호를 전송하고, 복호된 결과 수신된 패킷에 오류가 검출되는 경우 이를 소멸시키지 않고 저장한 뒤에 향후 재 전송되어 올 패킷과 연성결합(Soft Combining)을 하는 방식을 말한다. 이때 재 전송되는 패킷은 항상 동일한 부호가 사용된다. 따라서 상기 패킷 다이버시티 결합 방식은 랜덤 채널에서 일종의 심볼 에너지를 평균하는 과정(Symbol Averaging)으로 볼 수 있으며, 수신 심볼 연성출력(soft output)을 평균함으로써 얻는 잡음전력감소 효과와 페이딩 채널에서 복수개의 심볼을 전송함으로써 다중성 채널에서 제공되는 다중성 이득(diversity gain)만을 사용하는 방식이라고 볼 수 있다. 이러한 패킷 다이버시티 결합 방식의 효과에 반해서 패킷 코드 결합 방식은 이러한 패킷 다이버시티 결합 방식의 이득 이외에도 코드 구조(Code structure)에 따른 추가의 코딩 이득(Coding Gain)을 가지고 있다.

<19> 현재까지의 패킷통신 시스템에서는 구현의 용이함 때문에 대부분 패킷 다이버시티 결합 방식을 사용하고 있으며, 동기방식의 IS-2000 시스템과 비동기방식의 UMTS(Universal Mobile Telecommunication System) 시스템 등에서도 이러한 방식이 고려되고 있다. 이는 기존의 패킷통신 시스템들이 대부분 컨볼루션 부호(Convolutional codes)를 사용하였고, 부호율 R 이 낮은 컨볼루션 부호를 사용하는 경우 그리 큰 이득이 제공되지 않기 때문이었다. 즉, 부호율 $R=1/3$ 을 사용하는 시스템에서 재전송을 행할 때 패킷 다이버시티 결합 방식을 사용하는 경우와 패킷 부

호 결합 방식을 사용하여 1/6 부호율을 사용하는 경우의 성능 차이가 그리 크지 않기 때문에 구현의 복잡도를 고려하여 패킷 다이버시티 결합 방식을 사용한 것이다.

<20> 그러나, 오류정정부호(Forward Error Correction Codes: FEC)로 터보 부호(Turbo Codes)를 사용하는 경우에는 이러한 기존의 개념과는 다른 방식이 요구된다. 왜냐하면, 터보 부호는 반복적인 복호화(iterative decoding)에 의해서 그 성능이 Shannon limit에 근접하도록 설계된 오류정정부호이고, 기존의 콘볼루션 부호와는 달리 부호율에 따른 성능의 차이가 분명하게 존재하기 때문이다. 이러한 점에 착안할 때 터보 부호를 사용하는 통신시스템에서 패킷 부호 결합 방식을 구현한다면, 재전송시 시스템의 성능을 향상시킬 수 있을 것이다.

<21> 그러므로, 본 발명은 상기 패킷 부호 결합 방식과 패킷 다이버시티 결합 방식을 모두 고려하는 복합 재전송방식을 사용하는 시스템에서 최적의 패킷 부호 결합을 위한 패킷 구조를 구성하는 방법 및 장치를 제시하고자 하는 것임을 밝혀두는 바이다. 이러한 본 발명이 제시하는 복합 재전송 방식의 통신시스템은 데이터 레이트에 따라서 패킷 부호 결합 방식과 패킷 다이버시티 결합 방식을 다음과 같이 가변적으로 사용할 수 있다. 예를 들어, 부호율 $R=1/5$ 인 터보 부호를 사용하는 시스템에서 재전송된 패킷들로부터 연성 결합에 의해 구한 부호어(code word)의 전체적인 부호율(overall code rate)이 $1/5$ 이 되기까지는 패킷 부호 결합 방식을 사용하고, 이후부터 재전송되는 패킷들에 대해서는 우선 패킷 다이버시티 결합 방식을 사용하고 이어서 패킷 부호 결합 방식을 사용한다. 즉, 처음 전송하는 패킷의 부호율 R 이 $1/3$ 인 경우 다음 번 재전송 요구 때에는 전체적인 부호율 R 이 $1/5$ 이 되도록 나

머지 리던던시(redundancy)를 전송한다. 따라서 수신기가 두 개의 패킷들을 모두 수신하면 전체적인 부호율은 $R=1/5$ 이 되고, 그 이후에 전송되는 패킷들에 대해서 각 방식들을 반복해서 전송하면 수신기는 패킷 다이버시티 결합 방식을 사용한 후에 부호율 $R=1/5$ 을 기준으로 재전송 패킷에 대해서 패킷 부호 결합을 수행한다.

<22>

한편, 전송한 바와 같은 가변적인 혹은 고정적인 부호율을 사용하고 패킷 다이버시티 결합 방식을 사용하는 시스템 혹은 패킷 부호 결합 방식을 사용하는 시스템에서는, 각각의 부호율에 따른 부호어의 길이가 전송하고자 하는 물리채널의 프레임 길이보다 작거나 같아서 항상 하나의 물리채널의 전송 프레임을 수신하는 경우 하나의 부호어가 수신될 수 있도록 전송 패킷의 길이와 구조를 결정하는 것이 최적의 전송을 위한 필요 조건이 된다. 그러나 일반적으로 전송 프레임의 길이는 시스템의 물리채널 사양에 의해서 결정되는 파라미터이므로 이러한 구조를 만들기는 용이하지 않다. 따라서 기존에는 비효율적인 추가비트(padding bits) 등을 사용하여 이를 보상하는 방식이나, 전송 프레임의 길이와 부호어의 길이 비가 정수가 되지 않은 상태에서 연속적으로 부호어를 전송하고, 수신기가 다시 이를 전송 프레임 단위로 결합한 후에 부호어 단위로 재분리하는 방식 등이 고려되어 왔다. 그러나, 이러한 기존의 방식들을 사용하는 경우에는 부호어의 길이와 전송 프레임의 길이가 정확하게 일치하는 경우에 비하여 다음과 같은 문제점이 있다.

<23>

1. 각각의 부호율에 따른 부호어의 길이가 전송하고자 하는 물리채널의 프레임 길이보다 작은 경우에, 물리채널 전송프레임 내의 잔여분의 심볼 위치, 즉 잔여 심볼에 대부분 비효율적인 추가비트(padding bits) 등을 사용하여 이를 보상하는

방식이 가능하다. 그러나 이러한 방식은 추가 비트에 따른 시스템의 처리율(Throughput)을 감소시키는 문제점이 있다. 특히, 전송율이 작을수록 더욱 더 많은 전송율 손실이 발생하게 된다.

<24>

2. 각각의 부호율에 따른 부호어의 길이가 전송하고자 하는 물리채널의 프레임 길이보다 작은 경우에, 항상 하나의 물리채널의 전송 프레임에 하나의 부호어를 전송하고자 하는 경우 각각의 전송율에 따른 매우 다양한 부호율을 지닌 부호들이 요구된다. 그러나 대부분의 경우 전송에 사용되는 부호율들은 기본이 되는 모 부호(mother code)로부터 일정한 천공(puncturing)과 반복(repetition)에 의해서 구해지므로, 현실적으로 제공될 수 있는 부호율의 종류는 제한적일 수 밖에 없다. 게다가, 이러한 방식을 사용하더라도 하나의 전송프레임에 잔여 심볼이 발생하고, 이를 추가비트(padding bits)로 보상해야 하는 경우가 대부분이다. 따라서 이러한 방식의 경우에도 그 양은 감소하겠지만 추가비트에 따른 시스템의 처리율(Throughput)을 감소시키는 문제점이 있다.

<25>

3. 전송 프레임의 길이와 부호어의 길이 비가 정수가 되지 않은 상태에서 복수개의 전송 프레임을 직렬로 연결시킨 후에 잔여비트를 사용하지 않고 하나의 부호어가 전송된 직후 연속해서 다음의 부호어를 연결해서 전송하는 방식이 가능하다. 상기 방식이 사용되는 경우, 수신기는 전송 프레임 단위로 결합을 행한 후에 부호어 단위로 재 분리를 하게 된다. 그러나 이 경우에는 하나의 전송 프레임에 복수개의 부호어가 전송될 수 있다. 이는 동일한 전송 물리채널 파라미터를 가지는 전송 프레임에서 서로 다른 복수개의 부호어가 사용될 수 있음을 의미하므로,

이러한 상황에서 수신기가 서로 다른 복수개의 부호어를 구별해서 운영하기란 현실적으로 매우 어렵다. 즉, 하나의 전송 프레임에 복수개의 동일한 부호어가 전송될 수는 있으나, 복수개의 서로 다른 부호어가 전송되기는 매우 어렵다. 물론, 이러한 문제점은 하나의 전송 프레임내에서 부호어의 변화를 알리는 정보 심볼을 사용함으로써 해결될 수 있으나, 각각의 부호어를 저장하는 메모리가 요구되고 이 메모리를 제어하는 제어부가 필요하다는 등의 상당히 복잡한 구조가 요구된다.

<26> 2. Redundancy Selection

<27> 2-1. Quasi-Complementary Code Set

<28> 패킷 부호 결합(Packet code combining)을 사용하는 대표적인 시스템에는 Hybrid ARQ 시스템이 있다. 즉, HARQ Type I, Type II, Type III에서 모두 패킷 부호 결합(Packet Code Combining)을 사용할 수 있으며, 이러한 시스템에 앞서 설명한 기술을 적용하고자 본 발명에서는 준 보완 터보부호 셋(Quasi-complementary turbo code set)을 사용한다.

<29> 예를 들어, 패킷 전송을 위한 기본 단위의 정보비트 블록을 TU(transport unit)라고 정의하면 패킷 부호 결합(Packet code combining)을 사용하는 상기의 Hybrid ARQ 시스템 즉, HARQ Type I, Type II, Type III에서는 각각의 TU 전송 시마다 아래에 제시하는 준 보완 터보부호 셋(Quasi-complementary turbo code set) 중 하나의 서브 부호(sub code) C_i 가 할당된다.

<30> 물론 재전송의 기본단위는 초기 전송에 사용되는 TU의 크기와 동일한 것을 사용할 수도 있으며, 혹은 매 전송마다 다른 것을 사용할 수 도 있다. 단, 한번의 초기 전송과 각각의 재전송 시마다 아래에 주어지는 준 보완 터보부호 셋(Quasi-complementary turbo code set)을 사용한다.

<31> 준 보완 터보 부호(Quasi-complementary code) C_q 는 부호 셋 크기(code set size) S 를 가지며, 각각의 서브 부호(sub codes) C_i (여기서, $i=0,1,2,\dots,S-1$)를 조합함으로써 모 부호(mother code) C 를 복원(reconstruction)할 수 있거나 혹은 이 보다 작은 부호율을 가지는 새로운 C_q 를 구성(construction)할 수 있는 부호들을 의미한다. 여기서 모 부호라 함은 복호기(encoder)가 가질 수 있는 최소의 부호율을 의미한다. 이러한 모 부호를 사용하는 시스템에서 아래의 성질을 만족하는 경우 준 보완 터보 부호(Quasi-complementary turbo codes)라 한다.

<32> Original code C with code rate $R=R_m$ or a code C with code rate $R < R_m$

$$= \bigcup_{i=0}^{S-1} C_i$$

<33> (where S is number of sub codes with code rate of R_i)

<35> 아래에 준 보완 터보부호(quasi-complementary turbo codes)를 사용하고, 전송 단위를 TU로 정의하고 초기전송과 재전송에서 모두 동일한 TU 크기를 사용하는 시스템에서 구체적인 전송방식을 보였다. 물론 각각의 전송에서 상이한 TU 크기를 사용하는 시스템에서의 전송방식도 제안이 가능하다. 하지만, 후술되어질 설명의

편의를 위해서 모두 동일한 TU 크기를 사용하는 시스템에서의 구체적인 전송방식을 보였다. 아래의 예는 서브 부호들(sub codes)의 수가 $S=4$ 이고, 모 부호율(mother code rate)이 $R=1/5$ 인 경우를 예를 보인 것이다.

- <36> - 초기전송 및 재전송 단위는 TU 단위로 이루어지며, 각각의 초기 전송 및 재 전송 시에 준 보완 부호 셋(quasi-complementary code set)의 서브 부호 C_i 가 각각 전송된다.
- <37> - 전체 부호율(Overall code rate)이 $1/5$ 보다 큰 경우에는 재전송이 요청될 때마다 준 보완 부호 셋(quasi-complementary code set)의 서브 부호 C_i 가 각각 전송되며, 그 순서는 $C_0, C_1, C_2, \dots, C_{S-1}$ 가 된다. 이 과정은 패킷 부호 결합(Packet code combining)을 수행하는 단계를 의미한다.
- <38> - 재전송과 초기전송을 포함한 수신된 패킷의 모든 연성결합에 의한 전체 부호율(Overall code rate)이 $1/5$ 보다 작거나 같은 경우에는 재전송이 요청될 때마다 준 보완 부호 셋(quasi-complementary code set)의 서브 부호 C_i 가 각각 다시 반복되어 전송되며, 그 순서는 $C_0, C_1, C_2, \dots, C_{S-1}$ 이 된다. 이 과정은 패킷 다이버시티 결합(Packet diversity combining)을 수행하는 단계를 의미한다.
- <39> - 준 보완 부호 셋 크기(Quasi-complementary code set size)는 임의의 값을 사용할 수 있으며, 이는 초기에 결정되는 복호기의 모 부호율에 의해서 결정된다. 여기서는 모 부호율 $R=1/5$ 이고, $R=2/3$ 인 부호율을 재전송을 위한 서브 부호율로 사용하므로 최대 4개의 서브 부호가 생성 가능하다. 따라서 $S=4$ 를 사용한다.

<40>

하기의 <표 1>에서는 준 보완 터보부호 셋(Quasi-complementary turbo codes sets)과 이를 사용하는 순방향 트래픽 채널에서의 패킷 전송율(Forward Traffic Channel packet data rates)의 일 예를 보이고 있다.

【표 1】

<41>

| Set Size S | Code Set | Subcode Rate Set | Data Rates |
|------------|--|--|--|
| 1 | {C ₀ } | C ₀ : R ₀ =1/6 | 307.2kbps 153.6kbps 76.8kbps 38.4kbps 19.2kbps |
| 2 | {C ₀ , C ₁ } | C ₀ : R ₀ =1/3 C ₁ : R ₁ =1/3 | 1228.8kbps 921.6kbps 614.4kbps 307.2kbps |
| 4 | {C ₀ , C ₁ , C ₂ , C ₃ } | C ₀ : R ₀ =2/3 C ₁ : R ₁ =2/3 C ₂ : R ₂ =2/3 C ₃ : R ₃ =2/3 | 2457.6kbps 1843.2kbps 1228.8kbps |

<42>

상기의 <표 1>은 현재 IS-2000의 1XEV/DV에서 사용될 것으로 보이는 가용한 순방향채널의 전송율에 따라서 구한 것으로 모 부호율 R=1/5이고, 각각의 서브 부호의 부호율로 R=2/3, 1/3, 1/6을 사용하는 경우의 예를 보인 것이다. 상기 <표 1>에서 보여지고 있듯이 서브 부호율(sub code rate)이 R=1/6의 경우에는 모 부호율(Mother code rate) R=1/5보다 작으므로 매 전송 시마다 동일한 부호가 사용되며, 이를 C₀로 표시하였다. 반면에 서브 부호율(sub code rate)이 R=1/3의 경우에는 모 부호율(Mother code rate) R=1/5보다 크므로 매 전송 시마다 상이한 부호가 사용되며 이를 C₀, C₁로 표시하였다. 이 경우 셋 크기(Set Size)는 S=2가

된다. 상기 서브 부호율(Sub code rate)이 $R=2/3$ 의 경우에는 모 부호율(Mother code rate) $R=1/5$ 보다 크므로 매 전송 시마다 상이한 부호가 사용되며, 이를 C_0 , C_1 , C_2 , C_3 으로 표시하였다. 이 경우 셋 크기(Set Size)는 $S=4$ 가 된다. 결국 각각의 서브 부호들이 셋 크기(Set Size) S 만큼 전송되는 경우 수신기는 원래의 모 부호율(mother code rate) R 을 복원할 수 있으며, 복호기(encoder)가 제공하는 최대의 부호화 이득(coding gain)을 제공할 수 있다.

<43> 2-2. Puncturing Matrix for Quasi-complementary codes

<44> 하기의 <표 2>에서는 상기 <표 1>에서 사용하는 각각의 부호율에 따른 천공 매트릭스(puncturing matrix)의 일 예를 보여주고 있다.

【표 2】

| Code rates | C_0 | C_1 | C_2 | C_3 |
|------------|---|---|---|---|
| $R=1/6$ | $\begin{bmatrix} X \\ Y_0 \\ Y_1 \\ Y'_0 \\ Y'_1 \end{bmatrix} = \begin{bmatrix} 2 \\ 1 \\ 1 \\ 1 \\ 1 \end{bmatrix}$ | NA | NA | NA |
| $R=1/3$ | $\begin{bmatrix} 1 \\ 1 \\ 0 \\ 1 \\ 0 \end{bmatrix}$ | $\begin{bmatrix} 1 \\ 0 \\ 1 \\ 0 \\ 1 \end{bmatrix}$ | NA | NA |
| $R=2/3$ | $\begin{bmatrix} 0 \\ 1 \\ 0 \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} 1 \\ 0 \\ 0 \\ 1 \\ 0 \end{bmatrix}$ | $\begin{bmatrix} 1 \\ 0 \\ 0 \\ 1 \\ 0 \end{bmatrix} = \begin{bmatrix} 0 \\ 1 \\ 0 \\ 0 \\ 0 \end{bmatrix}$ | $\begin{bmatrix} 0 \\ 0 \\ 1 \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \\ 1 \end{bmatrix}$ | $\begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \\ 1 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 1 \\ 0 \\ 0 \end{bmatrix}$ |
| $R=$ | $\begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}$ | $\begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 \end{bmatrix}$ | $\begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 1 \end{bmatrix}$ | $\begin{bmatrix} 1 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix}$ |

<46> 상기의 <표 2>에서 보여지고 있듯이 R-1/5 터보부호를 모 부호로 사용하고, 각각의 전송마다 R-2/3 부호를 사용한다고 하면 4개의 정보 심볼들(information symbols)에 대해서 20개의 코드 심볼들(code symbols)이 발생된다. 이 중에서 14개를 천공(puncturing)하면 R-2/3 터보부호들(turbo codes)이 된다. 그러면 패킷 다이버시티 결합(packet diversity combining)은 전술한 천공 매트릭스(puncturing matrix)에 의해 생성되는 {C0}을 재전송이 요구될 때마다 반복해서 전송하고, 이를 연성 결합(soft combining)하는 것을 의미한다. 반면에 패킷 부호 결합(packet code combining)은 아래와 같이 재전송 요청이 있을 때마다 서로 다른 부호 심볼들(coded symbols)을 전송하고, {C0, C1, C2, C4} 모두 전송된 이후에는 앞서와 유사하게 패킷 다이버시티 결합(packet diversity combining)을 사용한다. 따라서 패킷 부호 결합(packet code combining)을 사용하는 HARQ Type III의 경우에는 4번의 전송이 이루어진 이후에 R-1/5의 전 코드 심볼들(full coded symbols)을 모두 가지고 복호(decoding)가 가능해 진다.

<47> 전술한 <표 2>에서 X, Y0, Y1, Y'0, Y'1은 각각 아래와 같이 R=1/5 터보 부호화기(Turbo encoder)에서 출력되는 부호 심볼들을 의미한다.

<48> X: Systematic code symbol (Information symbols)

<49> Y0: Redundancy symbol from the upper component encoder of Turbo encoder

<50> Y1: Redundancy symbol from the lower component encoder of Turbo encoder

<51> Y'0: Redundancy symbol from the upper component encoder of Turbo

encoder

<52> Y'1: Redundancy symbol from the lower component encoder of Turbo encoder

<53> 본 발명의 실시 예에 따른 구체적인 터보 부호화기(Turbo encoder)의 구조는 도 1에서 보여지고 있는 바와 같다.

<54> 상기 도 1을 참조하면, 제1구성부호기 201은 입력되는 정보비트열을 부호화하여 부호심볼 X , Y_0 , Y_1 을 출력한다. 인터리버 202는 상기 입력되는 정보비트열을 주어진 규칙에 의해 인터리빙하여 출력한다. 제2구성부호기 203은 상기 인터리버 202로부터 출력되는 상기 인터리빙된 정보비트열을 부호화하여 부호심볼 X' , Y'_0 , Y'_1 을 출력한다. 심볼 천공/반복기 204는 상기 제1 및 상기 제2구성부호기 201, 203으로부터의 출력 부호심볼들 X , Y_0 , Y_1 , X' , Y'_0 , Y'_1 을 입력하고, 제어부 205로부터의 제어에 의해 상기 부호심볼들을 천공 및 반복하여 출력한다.

<55> 상기 도 1에서의 X' 는 실제로 전송되지 않는다. 따라서, 부호율은 $R=1/5$ 이 된다. 전술한 <표 42>에서 '1'은 전송되는 심볼을 의미하며, '0'은 전송되지 않고 천공(puncturing)되는 심볼을 의미한다. 또한 '2'는 해당되는 심볼이 2번 반복되어 전송되는 것을 의미한다.

<56> 상기의 <표 2>에서 $R=1/6$ 의 경우 전송되는 부호 심볼의 수열은 " X , X , Y_0 , Y_1 , Y'_0 , Y'_1 , X , X , Y_0 , Y_1 , Y'_0 , Y'_1 ,"와 같다. 따라서 실제 전송되는 부

호 심볼의 수는 하나의 정보어 심볼(information symbol)에 대하여 6개의 부호어 심볼이 생성되므로 $R=1/6$ 이 된다. 상기 $R=1/6$ 의 천공 및 반복 매트릭스(Puncturing and Repetition Matrix)를 보면 복호기에서는 2번 반복된 X 심볼을 심볼연성결합(soft symbol combining)하여 하나의 심볼로 전환한 뒤에 복호를 수행한다. 따라서, 실제 복호기에 사용되는 부호율은 $R=1/5$ 로 간주된다. 이러한 정보어 심볼의 에너지가 증가된 $R=1/5$ 부호를 사용하는 경우에는 통상의 균일한 심볼 에너지를 사용하는 $R=1/5$ 부호에 비하여 성능이 개선된다. 또한 이러한 부호어 심볼을 반복에서 가장 적절한 위치는 바로 정보어 심볼이 된다. 이러한 관점에서 보면 $R=1/6$ 의 천공 및 반복 매트릭스(Puncturing and repetition matrix)를 사용함으로써 균일한 심볼 반복을 사용하고, 정보어 심볼을 반복하여 정보어 심볼의 에너지를 증가시키는 구조의 $R=1/6$ 부호를 사용할 수 있다.

<57> 상기한 <표 2>에서 $R=1/3$ 경우 전송되는 부호 심볼의 수열은 "X, Y0, Y'0, X, Y1, Y'1,"와 같다. 따라서 실제 전송되는 부호 심볼의 수는 하나의 정보어 심볼(information symbol)에 대하여 3의 부호어 심볼이 생성되므로 $R=1/3$ 된다. 그러나 각각의 전송에서 사용되는 C0, C1에서 사용한 천공 매트릭스(Puncturing matrix)가 다르므로 전송되는 심볼이 서로 상이하다는 것을 알 수 있다. 또한 C0과 C1을 연성결합하면 정보어 심볼인 X는 2번 반복되어 전송되고, 나머지 Y0, Y1, Y'0, Y'1은 각각 한번씩 전송된다. 따라서 상기 $R=1/6$ 인 경우와 동일하게 $R=1/5$ 의 복호기를 사용하여 복호가 가능하고 위에서 언급한 성질을 모두 만족하므로 성능을 보장 할 수 있다.

<58>

상기한 <표 2>에서 $R=2/3$ 첫 번째 경우 전송되는 부호 심볼의 수열은 "Y0, X, Y'0, X, Y'0, Y0, Y1, X, Y'1, X, Y'1, Y1,"와 같다. 따라서 실제 전송되는 부호 심볼의 수는 두개의 정보어 심볼(information symbol)에 대하여 3의 부호어 심볼이 생성되므로 $R=2/3$ 된다. 그러나 각각의 전송에서 사용되는 C0, C1, C2, C3에서 사용한 천공 매트릭스(Puncturing matrix)가 다르므로 전송되는 심볼이 서로 상이하다는 것을 알 수 있다. 또한 C0, C1, C2, C3을 연성결합하면 정보어 심볼인 X는 2번 반복되어 전송되고, 나머지 Y0, Y1, Y'0, Y'1은 각각 한번씩 전송된다. 따라서 상기 $R=1/6$ 인 경우와 동일하게 $R=1/5$ 의 복호기를 사용하여 복호가 가능하고 위에서 언급한 성질을 모두 만족하므로 성능을 보장 할 수 있다.

<59>

상기한 <표 2>에서 $R=2/3$ 두 번째 경우 전송되는 부호 심볼의 수열은 "X, Y0, X, X, Y'0, X, Y'0, Y0, Y'0, Y0, Y0, Y'0, Y1, Y1, Y'1, Y'1, Y1, Y'1, X, Y'1, X, X, Y1, X,"와 같다. 따라서, 실제 전송되는 부호 심볼의 수는 4개의 정보어 심볼(information symbol)에 대하여 6개의 부호어 심볼이 생성되므로 $R=2/3$ 된다. 그러나 각각의 전송에서 사용되는 C0, C1, C2, C3에서 사용한 천공 매트릭스(Puncturing matrix)가 다르므로 전송되는 심볼이 서로 상이하다는 것을 알 수 있다. 또한 C0, C1, C2, C3을 연성결합하면 정보어 심볼인 X는 2번 반복되어 전송되고, 나머지 Y0, Y1, Y'0, Y'1은 각각 한번씩 전송된다. 따라서 상기 $R=1/6$ 인 경우와 동일하게 $R=1/5$ 의 복호기를 사용하여 복호가 가능하고 위에서 언급한 성질을 모두 만족하므로 성능을 보장 할 수 있다.

<60> 3. 전송 패킷의 구조 및 전송방식

<61> 패킷이 전송되는 트래픽 채널에 HARQ III를 적용하는 방식에 있어 전송하고자 하는 각각의 패킷의 전송 프로토콜은 순방향 트래픽 채널과 역방향 트래픽 채널 모두에서 사용 가능하므로 여기서는 특별히 전달하고자 하는 의미의 혼돈 가능성이 없는 경우에 순방향 트래픽 채널과 역방향 트래픽 채널을 구별하지 않고 총칭하여 트래픽 채널로 명칭한다.

<62> 3-1. 전송패킷의 길이 및 물리채널 대응관계

<63> 패킷이 전송되는 트래픽 채널에 HARQ III를 적용하는 방식에 있어 전송하고자 하는 각각의 패킷의 길이는 가변적으로 결정할 수 있다. 여기서는 전송하고자 하는 하나의 패킷을 물리계층 패킷(PLP: Physical Layer Packet)으로 정의하고, 하나의 PLP에는 복수개의 서브 패킷이 존재할 수 있다. 이들을 각각 전송 유닛(TU: Transport Unit)이라고 명칭 한다. 상기 각각의 TU는 가변적인 크기를 가질 수 있다. 따라서 PLP 또한 가변적인 길이를 가질 수 있다. 물론 하나의 PLP에 한 개의 TU가 전달될 수도 있다. 따라서 아래에 기술되는 HARQ Type III의 프로토콜의 설명에서는 이를 구별하지 않고 대표되는 두 가지의 경우를 가지고 설명한다. 즉, 전송하고자 하는 패킷의 길이는 TU, 2TU, 3TU, 4TU를 가질 수 있고, 각각의 경우에 있어 최소의 TU당 비트수인 TU=768인 경우와 TU=1536인 경우 두 가지를 가진다. 여기서 TU=768인 경우를 이제부터 숏 포맷(Short Format)이라고 명칭하며, TU=1536인 경우를 롱 포맷(Long Format)이라고 명칭 한다. 따라서 아래에서는 숏 포맷(Short

Format)과 롱 포맷(Long Format)의 두 가지 경우로 크게 나누어서 구별하고, 각각의 경우에 1TU, 2TU, 3TU, 4TU에 해당하는 PLP 크기를 가지는 경우의 재전송 방식과 HARQ type III방식에 관해서 기술한다. 물론 하나의 PLP에 포함되는 TU의 최대 수는 가변적이며, 물리전송채널이 제공하는 전송율에 따라서 결정되는 값이며 여기서는 편의상 4를 사용한다.

<64>

한편, 하나의 PLP를 전송하는 물리채널의 전송단위는 물리전송채널의 최소 전송단위로 이루어진다. 여기서는 편의상 이를 슬롯(Slot)으로 정의하며, 이를 기준으로 하나의 PLP를 전송한다. 하나의 PLP의 전송을 위한 슬롯(Slot)의 수는 1에서 임의의 수까지 설정 가능하다. 하나의 슬롯(Slot)에 전송되는 데이터의 수는 가변적이며, 물리전송채널이 제공하는 전송율에 따라서 결정되는 값이다. 즉, 하나의 PLP에 대해서 해당되는 데이터 전송율을 감안하여 슬롯(Slot)의 수가 결정된다. 여기서는 그 중 한가지 예로서 하나의 PLP가 최소 1개의 슬롯(Slot)으로부터 최대 32개의 슬롯들(Slots)에 해당되는 숏 포맷(Short Format)의 경우와 하나의 PLP가 최소 2개의 슬롯(Slot)으로부터 최대 64개의 슬롯들(Slots)에 해당되는 롱 포맷(Long Format)의 경우를 고려한다. 이러한 구분은 앞서 TU=768과 TU=1536을 구분하는 것과 동일하며, TU=768이 최대 16개의 슬롯(slot)에 전송된다는 전제에서 이를 숏 포맷(Short Format)이라고 정의한다. 한편, 이의 두 배가되는 TU=1536 그리고 32 슬롯(slot)을 사용하는 것을 롱 포맷(Long Format)이라고 정의한다. 물론 이러한 구분은 전송하는 패킷의 길이에 따른 구분으로서 기본적으로 HARQ Type III의 프로토콜 동작 여부와는 큰 관계가 없으나 길이에 따른 시스템의 이득율(Throughput)의

변화와는 밀접한 관계가 있으므로 이를 구별하여 설명한다.

<65> 3-2. 트래픽 채널의 전송부호 선택 방식

<66> 트래픽 채널로 전송되는 PLP는 매 전송 시마다 Quasi-complementary code를 사용하여 부호화(encoding)된다. 즉, 트래픽 채널의 PLP 전송율(data rate)에 따라 하기의 <표 7>과 <표 8>로부터 설정되는 셋 크기(set size)가 S인 보완 부호 셋(complementary code set)에서 초기 전송에는 C0이 사용되고, 이후에 역방향 ACK 채널로부터 NACK가 전달될 때마다 트래픽 채널(Traffic Channel)로 C1, C2, CS-1, C0, C1, 의 순서로 전송되는 부호(code)가 순환(circulating)되어 선택된다.

<67> 3-3. 전송 트래픽 패킷의 구조 및 전송 방식

<68> 본 발명의 실시예에 따라 단일 슬롯(Single Slot) 혹은 복수개의 슬롯들(Multi Slots)을 사용하는 시스템에서, 하나의 PLP에 생성되는 부호어 심볼들은 하기에서 설명될 방식들에 의거 해당하는 1개의 슬롯 혹은 복수개의 슬롯들을 통해 전송될 수 있다. 본 발명의 실시예에 따라 트래픽 패킷을 전송하는 방식들을 설명하기 전에, 일반적으로 효율적인 패킷 전송을 위해서 고려되어야 하는 필요 조건들을 설명한다.

<69> (필요조건 1)

<70> 가변적인 혹은 고정적인 부호율을 사용하는 패킷 다이버시티 결합 방식을 사용하는 시스템 혹은 패킷 부호 결합 방식을 사용하는 시스템에서, 하나의 물리채널의 전송 프레임을 수신하는 경우 하나의 부호어가 수신될 수 있도록 전송 패킷의 길이와 구조를 결정하는 것은 재전송 방식을 사용하는 시스템에서 수신기의 구현 복잡도를 최소화하고, 송신기와의 전송 프로토콜을 단순화할 수 있다.

<71> (필요조건 2)

<72> 가변적인 혹은 고정적인 부호율을 사용하는 패킷 다이버시티 결합 방식을 사용하는 시스템 혹은 패킷 부호 결합 방식을 사용하는 시스템에서, 각각의 부호율에 따른 부호어의 길이와 전송을 위한 물리채널의 프레임 길이와의 차이를 최소화하는 것은 시스템의 처리율 (throughput)을 최대로 증가시킬 수 있다.

<73> (필요조건 3)

<74> 가변적인 혹은 고정적인 부호율을 사용하는 패킷 다이버시티 결합 방식을 사용하는 시스템 혹은 패킷 부호 결합 방식을 사용하는 시스템에서, 각각의 전송 부호어에 대한 ACK/NACK이 일정한 주기를 가지고 송신기에 전달될 수 있도록 주기성을 확보하는 것이 시스템의 처리율(throughput)을 증가시킨다.

<75> 상기와 같은 3가지 필요조건들을 모두 고려할 때 가장 좋은 방식은 항상 각

각의 부호율에 따른 부호어의 길이와 전송하고자 하는 물리채널의 프레임 길이가 동일하도록 하는 것이다. 그러나 각각의 부호율에 따른 부호어의 길이와 전송하고자 하는 물리채널의 프레임 길이가 동일하도록 전송패킷을 구성하려고 하는 경우, 현실적으로 너무나도 다양한 종류의 부호어가 요구되고, 수신기의 복잡도와 전송 프로토콜의 복잡도가 증가하게 된다. 따라서 본 발명은 하기에서 설명되는 바와 같은 방식들을 제안한다.

<76> (방식 1)

<77> 제1방식은 동일한 부호율을 사용하면서 실제 전송 프레임의 경계와는 관계없이 전송 가능한 모든 연속한 전송 프레임의 총 심볼 수(길이)를 하나의 부호어 심볼 수에 해당되는 길이로 분할하고, 이를 각각의 전송 프레임(슬롯)을 통해 전송하는 방식이다. 이러한 경우 각각의 슬롯에는 최대 두 가지의 서로 다른 부호어가 분할되어 혼합된 형태로 전송될 수 있다. 물론 이러한 경우 전송 가능한 모든 연속한 전송 프레임의 총 길이가 부호어 심볼 수의 정수배가 되지 않는다면, 최종 전송 프레임에는 사용하지 않는 여분의 심볼들이 존재할 수 있다. 이러한 여분의 심볼들은 수신기에서 심볼 연성 결합시 사용될 수도 있으나, 기본적으로 완전한 하나의 부호어를 전송하지는 못하게 된다. 이러한 제1방식을 위한 각 서브부호의 부호율 R_c 는 하기의 <수학식 1>과 같이 동일하게 결정된다.

【수학식 1】

<78>
$$Rc = (L \times RP) / (N - L)$$

<79> 상기 <수학식 A>에서, N은 전송 가능한 모든 연속한 전송 프레임의 총 심볼 수이고, L은 서브부호어에 대한 심볼 수이고, (N-L)은 최종 전송프레임에는 사용하지 않는 여분의 심볼 수이고, RP는 서브부호어의 반복 수이다.

<80> (방식 2)

<81> 제2방식은 상호 동일하거나 다른 부호율을 조합해 사용하면서 전송 프레임의 경계와 서브부호어 심볼의 경계가 일치하도록 하나의 PLP 전송에 사용되는 모든 연속한 전송 프레임의 총 심볼 수(길이)를 각 서브부호의 부호어심볼 수에 최대로 할당되도록 분할하고, 이를 각각의 전송 프레임(슬롯)을 사용하여 전송하는 방식이다. 이러한 제2방식에 따르면, 각각의 슬롯에는 항상 한 종류의 서브부호어가 전송된다. 이러한 제2방식에 있어서, 전송 가능한 모든 연속한 전송 프레임의 총 길이가 부호어심볼 수의 정수배가 되지 않는 경우에는 하기와 같은 (2-1)방식과 (2-2)방식으로 보완한다.

<82> (방식 2-1)

<83> (2-1)방식에 따르면, 하나의 서브부호에 할당된 전송프레임에 해당 서브부호어의 모든 심볼이 전송될 수 있도록 각 서브부호의 부호율이 결정된다. 이때 각 서

브부호의 부호율 R_c 는 각 서브부호의 부호율에 의해 발생하는 심볼 수가 각 전송프레임에 할당된 심볼수와 정확하게 일치하도록 하기의 <수학식 2>와 같이 설정된다. 이러한 (2-1)방식에 따르면, 사용하지 않는 여분의 심볼은 존재하지 않게 된다.

【수학식 2】

<84>
$$R_c = (L \times RP) / N$$

<85> 상기 <수학식 B>에서, N 은 전송 가능한 모든 연속한 전송 프레임의 총 심볼 수이고, L 은 서브부호어에 대한 심볼 수이고, $(N-L)$ 은 최종 전송프레임에는 사용하지 않는 여분의 심볼 수이고, RP 는 서브부호어의 반복 수이다.

<86> (방식 2-2)

<87> (2-2)방식에 따르면, 하나의 서브부호가 할당된 전송프레임에 전송될 수 있도록 각 서브부호의 부호율이 결정된다. 이때 각 서브부호의 부호율에 의해 발생하는 심볼 수는 각 전송프레임에 할당된 심볼 수보다 크도록 설정되고, 전송프레임에 배치되지 못한 나머지 심볼은 삭제(Pruning)된다. 각 서브부호의 부호율은 삭제되는 심볼의 수를 최소화하는 값중 최대의 정수비 부호율을 사용하는 것이 최적이다. 그러나 구현의 문제를 고려하여 적절한 정수비 부호율을 선택하여 사용하고 정합되지 않는 심볼 수만큼 삭제한다. 각 서브부호의 부호율 R_c 는 하기의 <수학식 3>과 같이 결정되고, 삭제되는 심볼 수(Number of Pruned Symbols) PS 는 하기의 <수학식 4>와 같이 결정된다.

【수학식 3】

<88>
$$Rc = (L \times RP) / N$$

【수학식 4】

<89>
$$PS = L - (N / RP)$$

<90> 상기 <수학식 3> 및 <수학식 4>에서, N은 전송 가능한 모든 연속한 전송 프레임의 총 심볼 수이고, L은 단일 부호어에 대한 심볼 수이고, (N-L)은 최종 전송 프레임에는 사용하지 않는 여분의 심볼 수이다.

<91> 한편, 상기 (2-1)방식과 (2-2)방식을 비교하면, N이 동일하다고 전제할 때, 즉 전송 가능한 모든 연속한 전송 프레임의 총 심볼 수 N이 동일하다고 전제할 때, 동일한 시간동안에 (2-2)방식은 (2-1)방식에 비해 더 낮은 부호율로의 전송이 가능하다. 따라서 부호화 이론(Coding Theory) 측면에서 (2-2)방식이 더 많은 부호이득(Coding Gain)을 가진다. 또한 (2-2)방식은 각 서브부호어의 부호율도 동일하게 가져갈 수 있는 장점이 있다. 또한 성능 측면에서도 (2-2)방식은 (2-1)방식보다 더 낮은 부호율을 사용하므로, 다소의 성능 이득이 제공되는 장점이 있다. 물론, 상기 (2-2)방식은 심볼 삭제(Pruning)에 의해 삭제될 심볼을 제어할 것이 요구된다. 그러나 심볼 삭제는 일정 시점 이후의 심볼의 전송을 중단시키는 것이므로, 단지 전송된 심볼 수를 누적하는 카운터(Counter)를 사용하여 손쉽게 구현할 수 있을 것이다.

<92> 전술한 바와 같은 본 발명의 실시예에 따른 트래픽 패킷 전송 방식들중에서

시스템의 특성에 따라서 어떠한 방식도 사용될 수 있다. 하기에서는 본 발명의 실시예에 따른 트래픽 패킷 전송 방식에 의해서 구성된 전송 패킷 구조가 구체적으로 설명될 것이다. 대표적으로, 하기에서는 상기 (2-2)방식에 의해서 구성된 전송패킷 구조가 <표 3> 내지 <표 8>과, 도 4 내지 도 20을 참조하여 구체적으로 설명될 것이다.

<93> 하기의 <표 3>과 <표 4>는 숏 포맷(the Short Format)(예: 단일 포맷, Single Format)을 사용하는 경우와 롱 포맷(the Long Format)(예: 2중 포맷, Double Format)을 사용하는 각각의 경우에 트래픽 채널 패킷 데이터 레이트에 따른 변조방식의 스펙트럼 효율도(modulation efficiency), 하나의 PLP에 할당되는 슬롯의 수(number of slots for 1 Physical Layer Packet), 하나의 PLP에 할당되는 비트 수(number of bits per 1 Physical Layer Packet), 부호율(code rate), 각각의 PLP 데이터 레이트에 대해서 하나의 TDM (Time Division Multiplexing)안에 존재하는 칩수(number of TDM chips in transmission duration for each Physical Layer Packet Data rates), 파일럿 칩수(Pilot chips), 프리앰블 심볼수(Preamble symbols), 데이터에 할당되는 칩수(Data chips), 16개의 왈시 부호(Walsh codes)를 기준으로 한 실제 가용한 왈시 부호 수와의 비(Available Walsh ratio), 하나의 PLP에서 가용한 변조심볼의 수(available modulation symbols for PLP data)를 나타낸다.

<94> 하기의 <표 5>과 <표 6>에 숏 포맷을 사용하는 경우와 롱 포맷을 사용하는 각각의 경우에 트래픽 채널 패킷 데이터 레이트에 따른 하나의 PLP에 할당되는 슬

롯의 수(number of slots for 1 Physical Layer Packet), 하나의 PLP에 할당되는 비트 수(number of bits per 1 Physical Layer Packet), 변조기가 하나의 PLP에 할당된 전체 슬롯에 대해서 제공하는 변조 심볼의 총 수(number of modulation symbols provided), 부호기가 생성하는 부호어 심볼에 필요한 변조 심볼의 수(number of modulation symbols needed), 하나의 PLP에 해당하는 부호어가 할당된 전체 슬롯을 통해 몇 번 반복되는가를 나타내는 수(number of pruned sequence transmission), 하나의 PLP에 해당하는 부호어 심볼 중에서 해당 슬롯에 전송되지 않고 제거되는 변조 심볼의 수(number of pruned modulation symbols), 실효 부호율(effective code rates)을 나타낸다.

<95> 하기의 <표 7>과 <표 8>은 슛 포맷을 사용하는 경우와 룡 포맷을 사용하는 각각의 경우에 트래픽 채널 패킷 데이터 레이트에 따라 삭제되는 변조 심볼의 수와 각각의 서브 부호어에서 삭제되는 심볼의 수를 각각 나타낸다. 상기 <표 7> 및 <표 8>에서 보듯이 각각의 데이터 레이트에 대해서 항상 일정한 삭제 심볼 수가 존재하므로 이를 기준으로 전송 패킷의 구조를 결정하면 된다.

<96> → <표 7>과 <표 8>에는 삭제되는 변조 심볼의 수와 서브 부호어에서 삭제되는 심볼의 수가 없음. 확인이 요망됨.

【표 3】

| Data Rates | Slots/ Physical Layer Packet | Bits in Physical Layer Packet | Available Walsh Ratio | Preamble Symbols | Pilot Chips | Data Chips | Available Data Modulation Symbols | TDM Chips |
|-------------|------------------------------------|--|-----------------------------|---------------------|----------------|---------------|--|--------------|
| 19.2kbps | 32 | 768 | 28/32 | 1024 | 8192 | 39936 | 34944 | 49152 |
| 38.4kbps | 16 | 768 | 28/32 | 1024 | 4096 | 19456 | 17024 | 24576 |
| 76.8kbps | 8 | 768 | 28/32 | 512 | 2048 | 9728 | 8512 | 12288 |
| 153.6kbps | 4 | 768 | 28/32 | 256 | 1024 | 4864 | 4256 | 6144 |
| 307.2kbps | 2 | 768 | 28/32 | 128 | 512 | 2432 | 2128 | 3072 |
| 614.4kbps | 1 | 768 | 28/32 | 64 | 256 | 1216 | 1064 | 1536 |
| 307.2kbps | 4 | 1,536 | 28/32 | 128 | 1024 | 4992 | 4368 | 6144 |
| 614.4kbps | 2 | 1,536 | 28/32 | 64 | 512 | 2496 | 2184 | 3072 |
| 1,228.8kbps | 1 | 1,536 | 28/32 | 64 | 256 | 1216 | 1064 | 1536 |
| 921.6kbps | 2 | 2,304 | 28/32 | 64 | 512 | 2496 | 2184 | 3072 |
| 1,843.2kbps | 1 | 2,304 | 28/32 | 64 | 256 | 1216 | 1064 | 1536 |
| 1,228.8kbps | 2 | 3,072 | 28/32 | 64 | 512 | 2496 | 2184 | 3072 |
| 2,457.6kbps | 1 | 3,072 | 28/32 | 64 | 256 | 1216 | 1064 | 1536 |

<97>

【표 4】

| Data Rates | Slots/ Physical Layer Packet | Bits in Physical Layer Packet | Available Walsh Ratio | Preamble Symbols | Pilot Chips | Data Chips | Available Data Modulation Symbols | TDM Chips |
|-------------|------------------------------------|--|-----------------------------|---------------------|----------------|---------------|--|--------------|
| 19.2kbps | 64 | 1,536 | 28/32 | 1024 | 16,384 | 80,896 | 70784 | 98,304 |
| 38.4kbps | 32 | 1,536 | 28/32 | 1024 | 8,192 | 39,936 | 34944 | 49,152 |
| 76.8kbps | 16 | 1,536 | 28/32 | 512 | 4,096 | 19,968 | 17472 | 24,576 |
| 153.6kbps | 8 | 1,536 | 28/32 | 256 | 2,048 | 9,984 | 8736 | 12,288 |
| 307.2kbps | 4 | 1,536 | 28/32 | 128 | 1,024 | 4,992 | 4368 | 6,144 |
| 614.4kbps | 2 | 1,536 | 28/32 | 64 | 512 | 2,496 | 2184 | 3,072 |
| 307.2kbps | 8 | 3,072 | 28/32 | 128 | 2,048 | 10,112 | 8848 | 12,288 |
| 614.4kbps | 4 | 3,072 | 28/32 | 64 | 1,024 | 5,056 | 4424 | 6,144 |
| 1,228.8kbps | 2 | 3,072 | 28/32 | 64 | 512 | 2,496 | 2184 | 3,072 |
| 921.6kbps | 4 | 4,608 | 28/32 | 64 | 1,024 | 5,056 | 4,424 | 6,144 |
| 1,843.2kbps | 2 | 4,608 | 28/32 | 64 | 512 | 2,496 | 2,184 | 3,072 |
| 1,228.8kbps | 4 | 6,144 | 28/32 | 64 | 1,024 | 5,056 | 4,424 | 6,144 |
| 2,457.6kbps | 2 | 6,144 | 28/32 | 64 | 512 | 2,496 | 2,184 | 3,072 |

<98>

【표 5】

| Data Rates (kbps) | Slots/ Physical Layer Packet | Bits in Physical Layer Packet | Number of Modulation Symbols Provided | Number of Modulation Symbols Needed | Number of Pruned Modulation Symbols | Number of Pruned Sequence Transmission | Code Rate | Effective Code Rate |
|-------------------|------------------------------|-------------------------------|---------------------------------------|-------------------------------------|-------------------------------------|--|-----------|---------------------|
| 19.2 | 32 | 768 | 2,304 | 34,944 | 1,920 | 16 | 1/6 | 0.176 |
| 38.4 | 16 | 768 | 2,304 | 17,024 | 1,408 | 8 | 1/6 | 0.180 |
| 76.8 | 8 | 768 | 2,304 | 8,512 | 704 | 4 | 1/6 | 0.180 |
| 153.6 | 4 | 768 | 2,304 | 4,256 | 352 | 2 | 1/6 | 0.180 |
| 307.2 | 2 | 768 | 2,304 | 2,128 | 176 | 1 | 1/6 | 0.180 |
| 614.4 | 1 | 768 | 1,152 | 1,064 | 88 | 1 | 1/3 | 0.361 |
| 307.2 | 4 | 1,536 | 2,304 | 4,368 | 240 | 2 | 1/3 | 0.352 |
| 614.4 | 2 | 1,536 | 2,304 | 2,184 | 120 | 1 | 1/3 | 0.352 |
| 1,228.8 | 1 | 1,536 | 1,152 | 1,064 | 88 | 1 | 2/3 | 0.722 |
| 921.6 | 2 | 2,304 | 2,304 | 2,184 | 120 | 1 | 1/3 | 0.352 |
| 1,843.2 | 1 | 2,304 | 1,152 | 1,064 | 88 | 1 | 2/3 | 0.722 |
| 1,228.8 | 2 | 3,072 | 2,304 | 2,184 | 120 | 1 | 1/3 | 0.352 |
| 2,457.6 | 1 | 3,072 | 1,152 | 1,064 | 88 | 1 | 2/3 | 0.722 |

<99>

【표 6】

| Data Rates (kbps) | Slots/ Physical Layer Packet | Bits in Physical Layer Packet | Number of Modulation Symbols Provided | Number of Modulation Symbols Needed | Number of Pruned Modulation Symbols | Number of Pruned Sequence Transmission | Code Rate | Effective Code Rate |
|-------------------|------------------------------|-------------------------------|---------------------------------------|-------------------------------------|-------------------------------------|--|-----------|---------------------|
| 19.2 | 64 | 1,536 | 4,608 | 70,784 | 2,944 | 16 | 1/6 | 0.174 |
| 38.4 | 32 | 1,536 | 4,608 | 34,944 | 1,920 | 8 | 1/6 | 0.176 |
| 76.8 | 16 | 1,536 | 4,608 | 17,472 | 960 | 4 | 1/6 | 0.176 |
| 153.6 | 8 | 1,536 | 4,608 | 8,736 | 480 | 2 | 1/6 | 0.176 |
| 307.2 | 4 | 1,536 | 4,608 | 4,368 | 240 | 1 | 1/6 | 0.176 |
| 614.4 | 2 | 1,536 | 2,304 | 2,184 | 120 | 1 | 1/3 | 0.352 |
| 307.2 | 8 | 3,072 | 4,608 | 8,848 | 368 | 2 | 1/3 | 0.347 |
| 614.4 | 4 | 3,072 | 4,608 | 4,424 | 184 | 1 | 1/3 | 0.347 |
| 1,228.8 | 2 | 3,072 | 2,304 | 2,184 | 120 | 1 | 2/3 | 0.703 |
| 921.6 | 4 | 4,608 | 4,608 | 4,424 | 184 | 1 | 1/3 | 0.347 |
| 1,843.2 | 2 | 4,608 | 2,304 | 2,184 | 120 | 1 | 2/3 | 0.703 |
| 1,228.8 | 4 | 6,144 | 4,608 | 4,424 | 184 | 1 | 1/3 | 0.347 |
| 2,457.6 | 2 | 6,144 | 2,304 | 2,184 | 120 | 1 | 2/3 | 0.703 |

<100>

【表 7】

<101>

| Data Rates | Slots/ Physical Layer Packet | Transport Unit/ Physical Layer Packet | Code rate | Repetition | ACK/NACK Period | ACK Indicator bits /Reverse ACK channel | Quasi complementary code set |
|--------------------|---------------------------------------|---|--------------|------------|--------------------|--|--|
| 19.2kbps | 32 | 1 | 1/6 | 16 | 2×4 slots | 1 | {C ₀ } |
| 38.4kbps | 16 | 1 | 1/6 | 8 | 2×4 slots | 1 | {C ₀ } |
| 76.6kbps | 8 | 1 | 1/6 | 4 | 2×4 slots | 1 | {C ₀ } |
| 153.6kbps | 4 | 1 | 1/6 | 2 | 2×4 slots | 1 | {C ₀ } |
| 307.2kbps | 2 | 1 | 1/6 | 1 | 2×4 slots | 1 | {C ₀ } |
| 614.4kbps | 1 | 1 | 1/3 | 1 | 1×4 slots | 1 | {C ₀ , C ₁ } |
| 307.2kbps | 4 | 2 | 1/3 | 2 | 2×4 slots | 2 | {C ₀ , C ₁ } |
| 614.4kbps | 2 | 2 | 1/3 | 1 | 2×4 slots | 2 | {C ₀ , C ₁ } |
| 1228.8kbps | 1 | 2 | 2/3 | 1 | 1×4 slots | 2 | {C ₀ , C ₁ , C ₂ , C ₃ } |
| 921.6kbps | 2 | 3 | 1/3 | 1 | 2×4 slots | 3 | {C ₀ , C ₁ } |
| 1843.2kbps mmmm | 1 | 3 | 2/3 | 1 | 1×4 slots | 3 | {C ₀ , C ₁ , C ₂ , C ₃ } |
| 1228.8kbps | 2 | 4 | 1/3 | 1 | 2×4 slots | 4 | {C ₀ , C ₁ } |
| 2457.6kbps | 1 | 4 | 2/3 | 1 | 1×4 slots | 4 | {C ₀ , C ₁ , C ₂ , C ₃ } |

【表 8】

<102>

| Data Rates | Slots/ Physical Layer Packet | Transport Unit/ Physical Layer Packet | Code rate | Repetition | ACK/NACK Period | ACK Indicator bits /Reverse ACK channel | Quasi complementary code set |
|------------|---------------------------------------|---|--------------|------------|--------------------|--|--|
| 19.2kbps | 32 | 1 | 1/6 | 16 | 2×8 slots | 1 | {C ₀ } |
| 38.4kbps | 16 | 1 | 1/6 | 8 | 2×8 slots | 1 | {C ₀ } |
| 76.6kbps | 8 | 1 | 1/6 | 4 | 2×8 slots | 1 | {C ₀ } |
| 153.6kbps | 4 | 1 | 1/6 | 2 | 2×8 slots | 1 | {C ₀ } |
| 307.2kbps | 2 | 1 | 1/6 | 1 | 2×8 slots | 1 | {C ₀ } |
| 614.4kbps | 1 | 1 | 1/3 | 1 | 1×8 slots | 1 | {C ₀ , C ₁ } |
| 307.2kbps | 4 | 2 | 1/3 | 2 | 2×8 slots | 2 | {C ₀ , C ₁ } |
| 614.4kbps | 2 | 2 | 1/3 | 1 | 2×8 slots | 2 | {C ₀ , C ₁ } |
| 1228.8kbps | 1 | 2 | 2/3 | 1 | 1×8 slots | 2 | {C ₀ , C ₁ , C ₂ , C ₃ } |
| 921.6kbps | 2 | 3 | 1/3 | 1 | 2×8 slots | 3 | {C ₀ , C ₁ } |

| | | | | | | | |
|------------|---|---|-----|---|-----------|---|--|
| 1843.2kbps | 1 | 3 | 2/3 | 1 | 1×8 slots | 3 | {C ₀ , C ₁ , C ₂ , C ₃ } |
| 1228.8kbps | 2 | 4 | 1/3 | 1 | 2×8 slots | 4 | {C ₀ , C ₁ } |
| 2457.6kbps | 1 | 4 | 2/3 | 1 | 1×8 slots | 4 | {C ₀ , C ₁ , C ₂ , C ₃ } |

<103> 도 4 내지 도 20은 본 발명의 실시예에 따른 각 데이터 레이트별 전송 패킷의 구조를 보여주는 도면이다.

<104> 상기 도 4 내지 도 20을 참조하면, 각 블록의 맨 처음에는 각 전송 PLP를 구분하기 위한 프리앰블(preamble)이 위치한다. 상기 프리앰블에 후속하는 블록이 변조 심볼들인데, 이는 부호심볼을 변조하여 구한 심볼들이며 각각의 수치는 해당 심볼들의 수이다. 도면에서 빗금으로 표시된 변조심볼들은 실제 물리채널로 전송되지 않고 삭제(pruning)되는 변조심볼들이고, 각각의 데이터 레이트에 따라서 다른 값이 사용된다. 그리고 도 7과 같이 복수개의 슬롯들을 사용하여 하나의 PLP를 전송하는 경우 부호어 단위의 반복이 이루어지며, 처음에 전송되는 준 보완 부호(Quasi Complementary Code)의 삭제심볼 수를 제외한 나머지 반복되는 부호어의 삭제심볼 수는 항상 일정하다.

<105> 3-3. 전송심볼 삭제(Symbol Pruning)

<106> 전술한 (2-2)방식에 따르면 심볼 삭제가 요구되는데, 이에 따른 동작을 도 21을 참조하여 구체적으로 설명한다.

<107> 도 21은 본 발명의 실시예에 따른 동작이 적용되는 시스템의 구성을 보여주는 도면이다.

<108>

상기 도 21에서, 터보 부호기(Turbo Encoder) 101은 전송을 위한 정보를 터보 부호에 의해 부호화하고, 부호어 심볼을 출력한다. 일례로, 상기 터보 부호기 101은 전송을 위한 정보를 부호율 $R=1/5$ 로 부호화한다. 부호 선택부(Code Selection) 102는 수신측으로부터의 재전송 요청시 상기 부호어 심볼과 다른 부호, 일종의 리던던시(redundancy)를 추가한다. 상기 터보 부호기 101과 상기 부호 선택부 102는 전송한 바와 같은 준 보완 부호를 발생하는 준 보완 부호 발생기로서 동작한다. 채널 인터리버(Channel Interleaver) 103은 상기 부호 선택부 102의 출력을 채널 인터리빙하여 채널 인터리빙된 심볼을 출력한다. 이러한 채널 인터리버 103으로는 블록 인터리버(block interleaver)가 사용될 수 있다. 변조기 104는 상기 채널 인터리버 103의 출력을 변조한다. 상기 변조기 104는 상기 채널 인터리버 103의 출력을 QPSK/8-PSK/16-QAM 변조할 수 있다. 심볼 삭제기 105는 상기 변조기 104의 출력 심볼들중 후반부의 일부 심볼들을 삭제한다. 상기 심볼 삭제기 105에 의해 삭제되는 심볼에 대해서는 후술될 것이다.

<109>

상기 도 21을 참조하면, 심볼 삭제(Symbol Pruning)의 동작은 채널 인터리빙 이후에 이루어진다. 물론 변조기 104가 채널 인터리버 103과 심볼 삭제기 105의 사이에 위치하지만, 상기 변조기 103은 채널 인터리버 103으로부터 출력되는 심볼열을 순서적으로 나누어서 변조 심볼에 대응시키는 기능 블록이므로, 상기 변조기 103은 채널 인터리버 103으로부터 전달되는 심볼의 순서와 동일한 순서로 변조 심볼을 출력한다. 따라서 도 4 내지 도 20에서 설명하는 심볼삭제를 사용하면, 결국 채널인터리버 103의 출력심볼들중 후반부 심볼들을 삭제하는 것과 동일하다. 이와

같이 채널 인터리버 103의 후반부가 삭제되면, 수신기에서 다음과 같은 현상이 발생된다.

<110> 예를 들어, 채널 인터리버의 크기가 NC이고, 심볼이 삭제된 부호심볼의 수가 $LC(=NC-PC)$ 이고, 삭제된 심볼 수가 PC라고 가정하면, PC개의 부호심볼들은 수신기에서 채널 디인터리빙에 의해서 랜덤하게 천공(Puncturing)된 것으로 간주된다. 상기 천공된 것으로 간주된 부분은 이레이저 심볼(Erasure symbol)로 처리된 후 복호가 수행한다. 이때 채널 디인터리버의 기능상 채널 인터리버의 후반부에 있는 심볼들은 채널 디인터리빙 이후에 랜덤하게 전체 부호심볼들에 배분되므로, 결국 복호기 입장에서는 PS개의 랜덤 오류가 발생된 것으로 간주되며 성능상으로 랜덤오류에 의한 성능감쇄만이 발생한다. 이러한 성질은 대부분의 시스템이 사용하는 부호가 랜덤오류정정부호라는 점을 감안할 때 성능상 문제가 없음을 의미한다. 또한 이 오류부분을 이레이저로 처리하면 실질적인 성능감쇄는 PC만큼 부호심볼이 감쇄된 새로운 부호율의 부호로 간주된다. 즉, 실질 부호율 Re 가 $NC/(NC-PC)$ 만큼 증가된 것으로 나타나고 이에 따른 성능이 얻어진다. 따라서 채널 인터리버의 후반부 심볼들을 삭제하더라도 ($Re < 1$)인 경우에 성능은 이 부호율에 해당되는 평균 성능을 제공할 수 있으므로, 문제가 되지 않는다. 이와 같은 성질의 채널 인터리버 중에서 가장 좋은 구조는 채널 인터리버의 후반부에 있는 심볼들이 채널 디인터리빙 이후에 균일한 간격으로 전체 부호심볼들에 배분되는 구조이며 이런 채널인터리빙을 사용하는 것이 최적의 방식이 된다.

<111> 전술한 본 발명의 상세한 설명에서는 구체적인 실시 예에 관해

설명하였으나, 본 발명의 범위에서 벗어나지 않는 한도 내에서 여러 가지 변형이 가능함은 물론이다. 그러므로 본 발명의 범위는 설명된 실시 예에 국한되어 정해져서는 안되며 후술하는 특허청구의 범위뿐만 아니라 이 특허청구의 범위와 균등한 것들에 의해 정해져야 한다.

【발명의 효과】

<112>

상술한 바와 같이 본 발명은 터보 부호를 사용하고 패킷 부호 결합 방식과 패킷 다이버시티 결합 방식을 함께 사용하는 복합 재전송형식의 통신시스템에서 전송 패킷 구조와 물리채널의 기능블록 전송 구조를 제안함으로써 패킷 부호 결합을 효율적으로 사용하고, 전송 효율을 극대화할 수 있는 이점이 있다.

【특허청구범위】

【청구항 1】

복합 재전송형식을 사용하는 통신시스템에서 전송 효율을 극대화하기 위한 패킷 전송 장치에 있어서,

전송을 위한 정보 비트들을 터보 부호에 의해 부호화하는 터보 부호화기와,

상기 터보 부호화기의 출력을 채널 인터리빙하는 채널 인터리버와,

상기 채널 인터리버의 출력을 미리 결정된 변조 방식에 의해 변조하는 변조기와,

상기 변조기에 의한 변조 심볼들중 각 서브부호의 부호율에 의해 발생하는 심볼 수가 각 전송 프레임에 할당된 심볼 수보다 크도록 설정하고, 전송프레임에 배치되지 못한 나머지 심볼을 삭제하는 심볼 삭제기를 포함함을 특징으로 하는 상기 장치.

【청구항 2】

복합 재전송형식을 사용하는 통신시스템에서 전송 효율을 극대화하기 위한 패킷 전송 방법에 있어서,

전송 가능한 모든 연속한 전송 프레임의 총 심볼 수를 하나의 부호어 심볼 수에 해당하는 길이로 분할하는 과정과,

상기 분할된 프레임을 각각의 전송 프레임을 사용하여 전송하는 과정을 포함

함을 특징으로 하는 상기 방법.

【청구항 3】

복합 재전송형식을 사용하는 통신시스템에서 전송 효율을 극대화하기 위한 패킷 전송 방법에 있어서,

하나의 물리계층 패킷(PLP) 전송에 사용되는 모든 연속한 전송 프레임의 총 심볼 수를 각각의 서브 부호의 부호어 심볼 수에 최대한 할당되도록 분할하는 과정과,

각 서브 부호의 부호율에 의해 발생하는 심볼 수가 각 전송 프레임에 할당된 심볼 수와 일치하도록 부호율을 조정하여 전송하는 과정을 포함함을 특징으로 하는 상기 방법.

【청구항 4】

복합 재전송형식을 사용하는 통신시스템에서 전송 효율을 극대화하기 위한 패킷 전송 방법에 있어서,

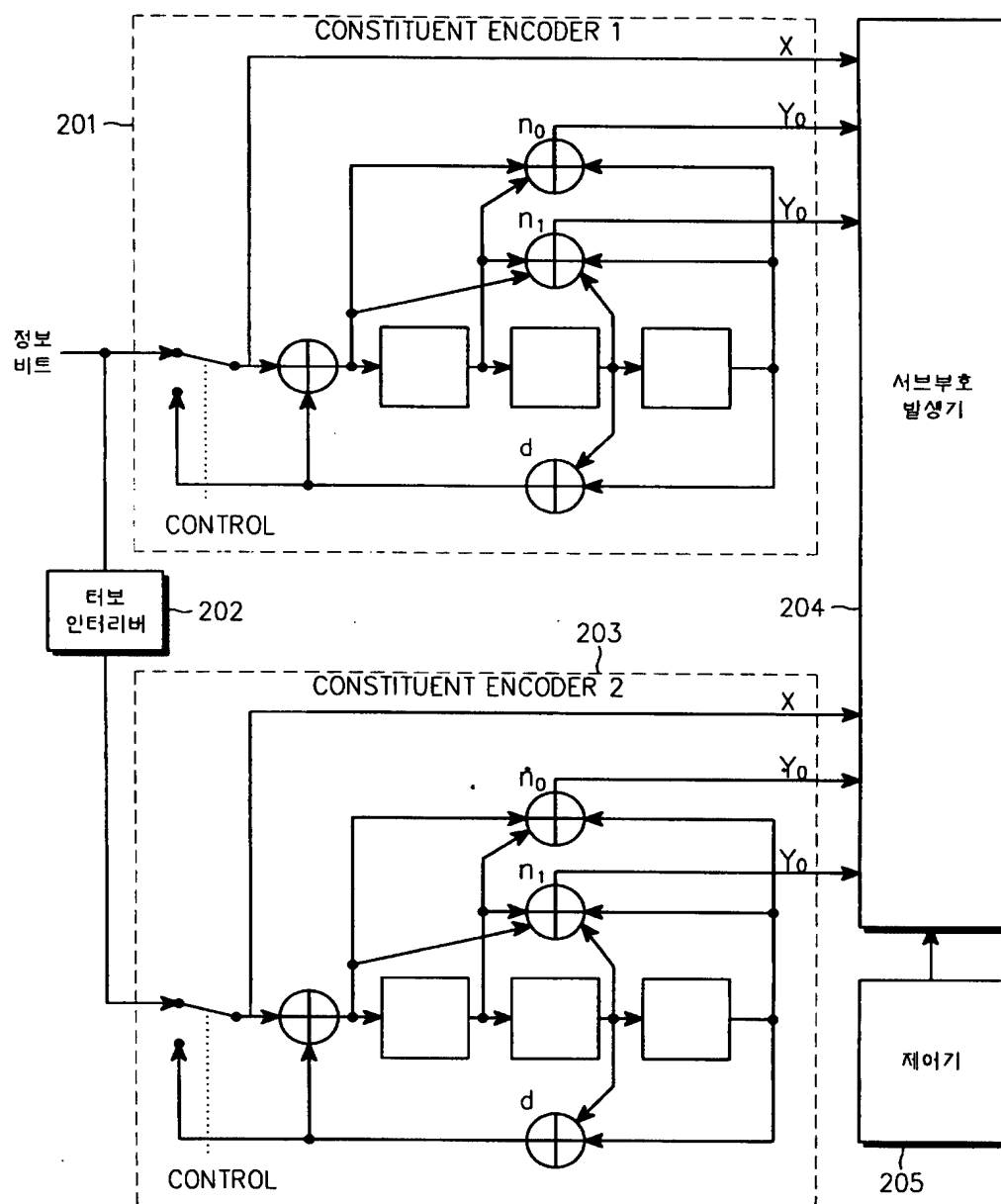
하나의 물리계층 패킷(PLP) 전송에 사용되는 모든 연속한 전송 프레임의 총 심볼 수를 각각의 서브 부호의 부호어 심볼 수에 최대한 할당되도록 분할하는 과정과,

각 서브 부호의 부호율에 의해 발생하는 심볼 수가 각 전송 프레임에 할당된

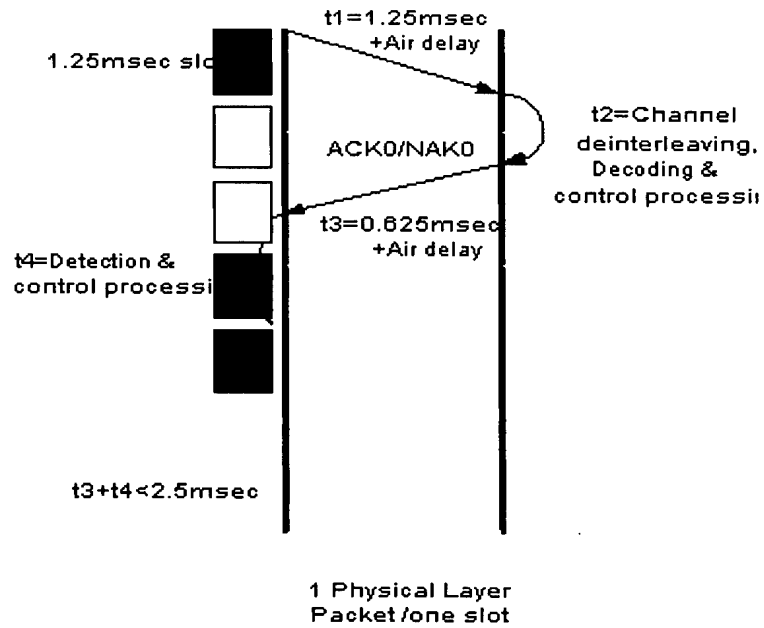
심볼 수보다 크도록 설정하고, 전송 프레임에 배치되지 못한 나머지 심볼을 삭제하여 전송하는 과정을 포함함을 특징으로 하는 상기 방법.

【도면】

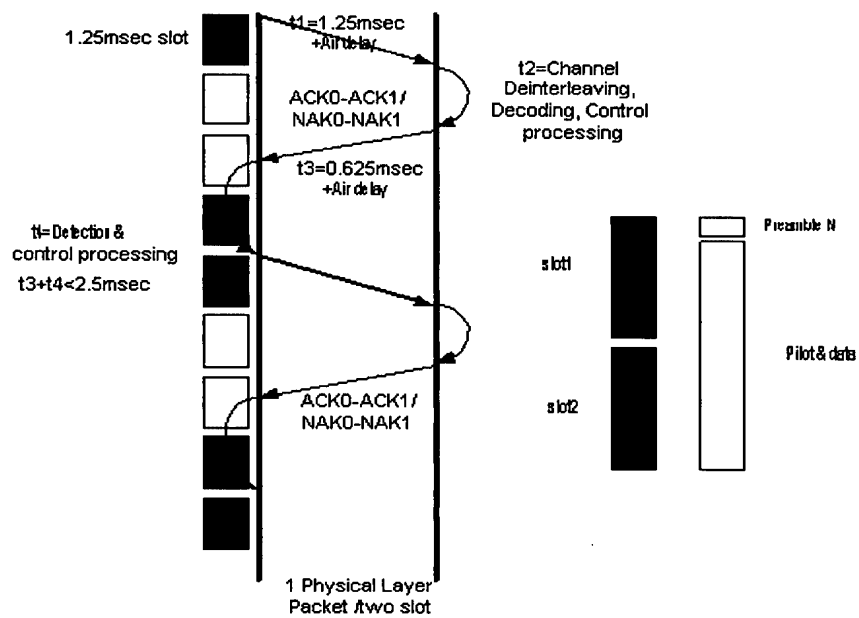
【도 1】



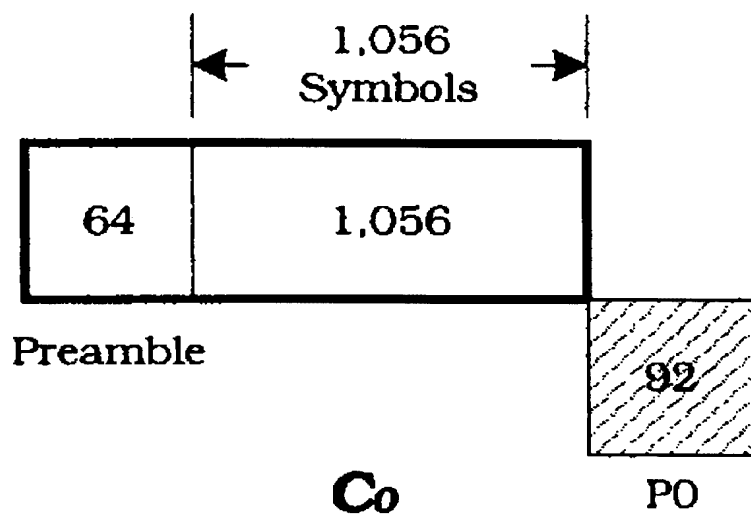
【도 2】



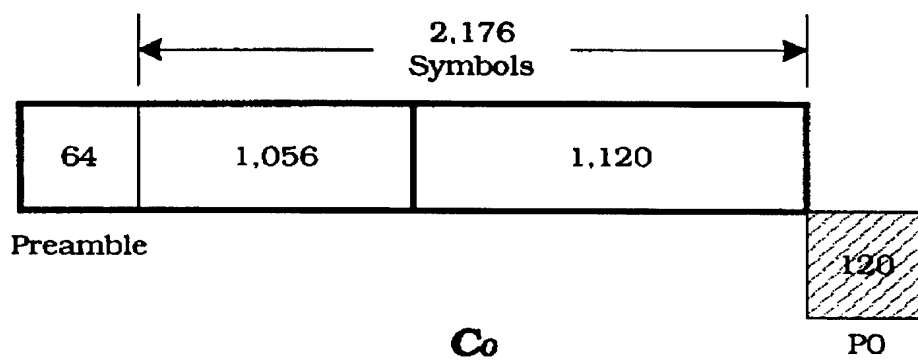
【도 3】



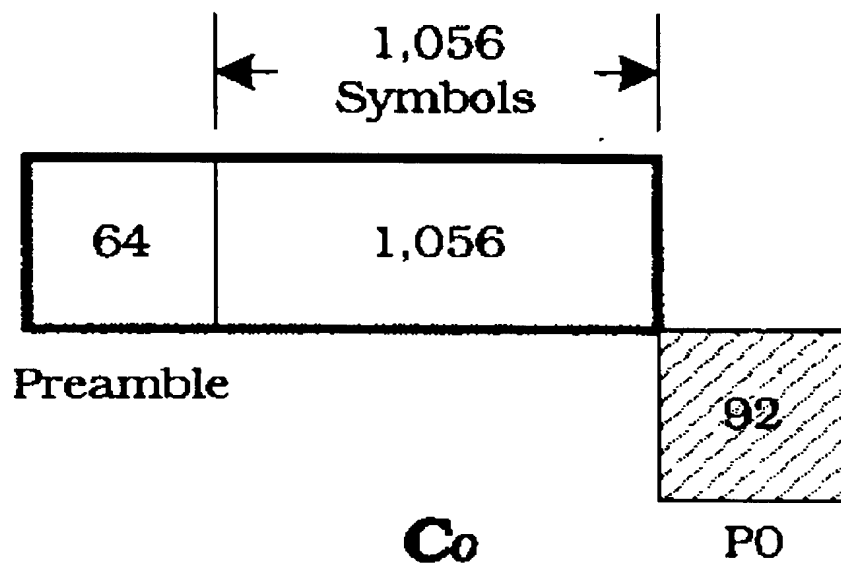
【도 4】



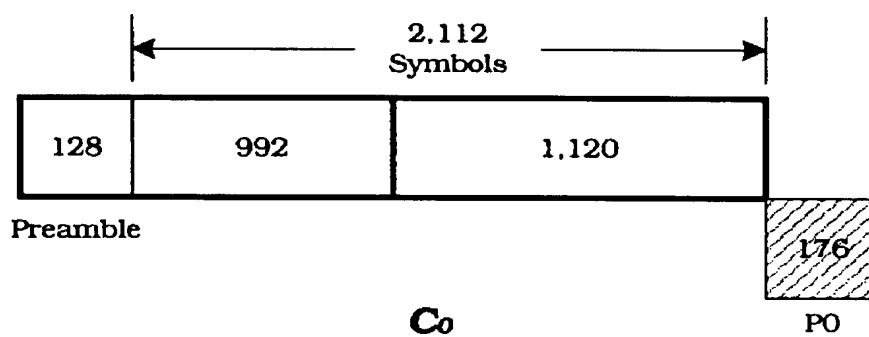
【도 5】



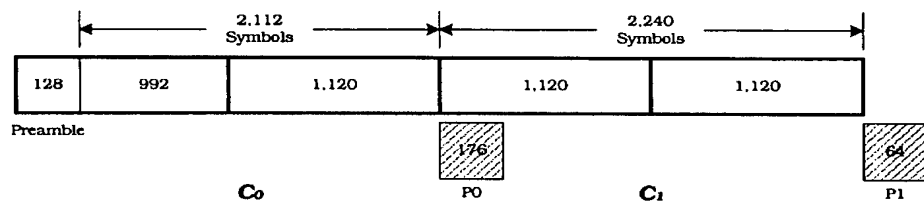
【도 6】



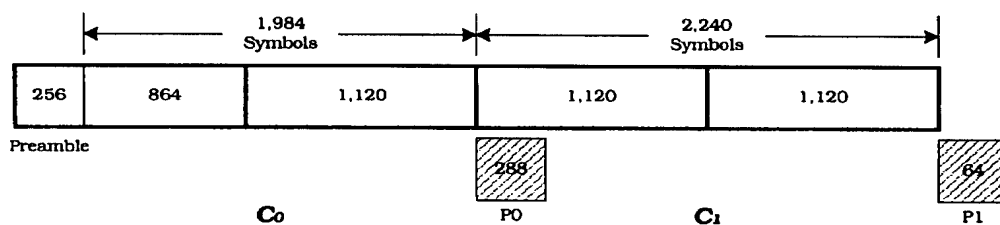
【도 7】



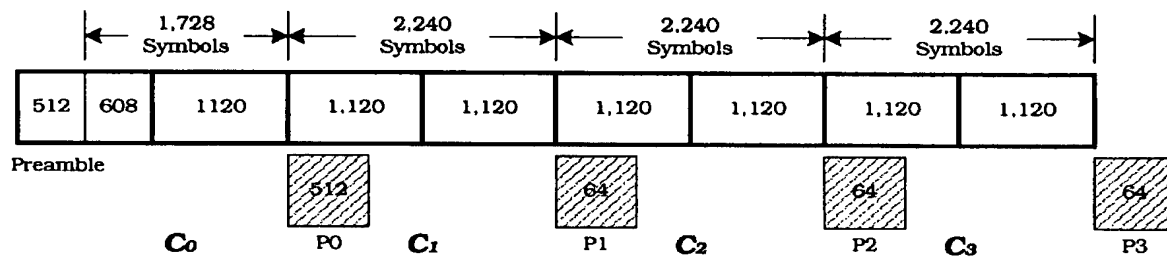
【도 8】



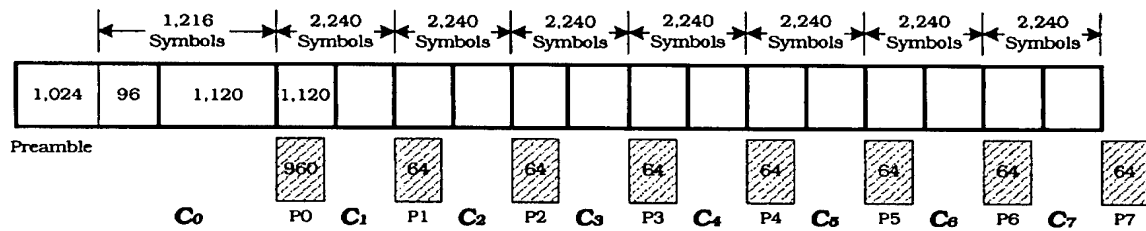
【도 9】



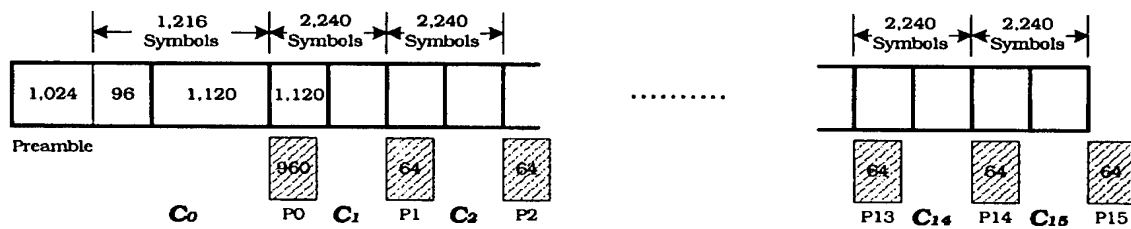
【도 10】



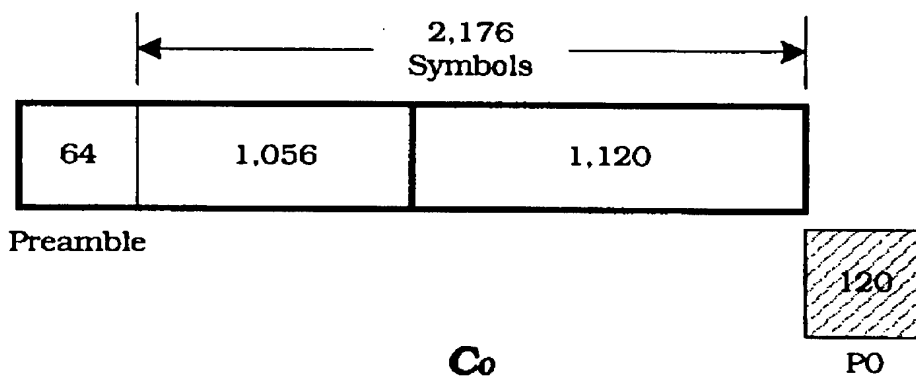
【도 11】



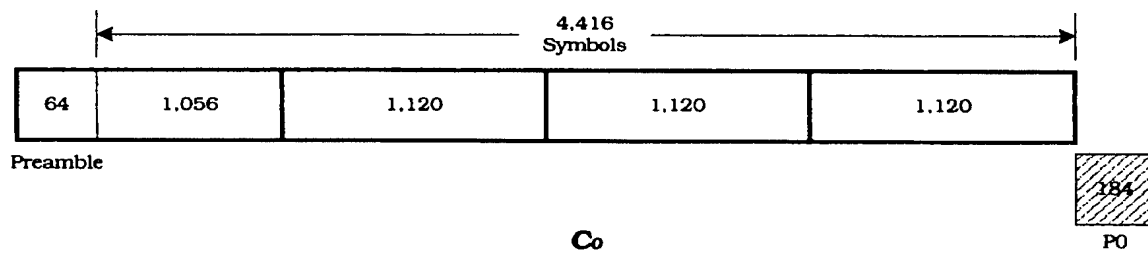
【도 12】



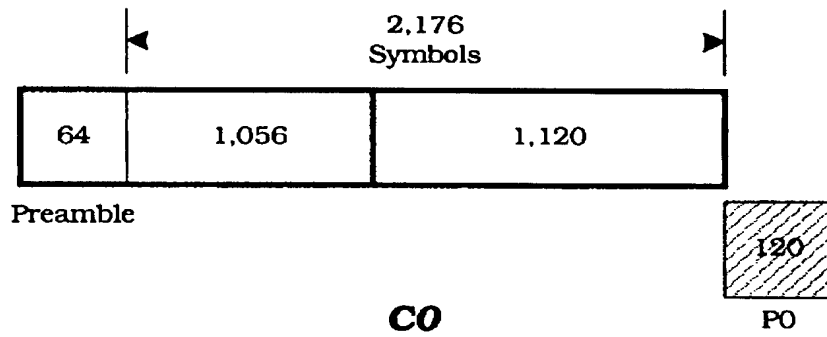
【도 13】



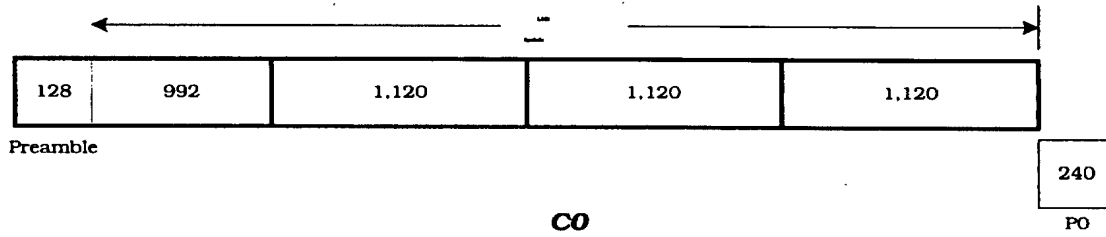
【도 14】



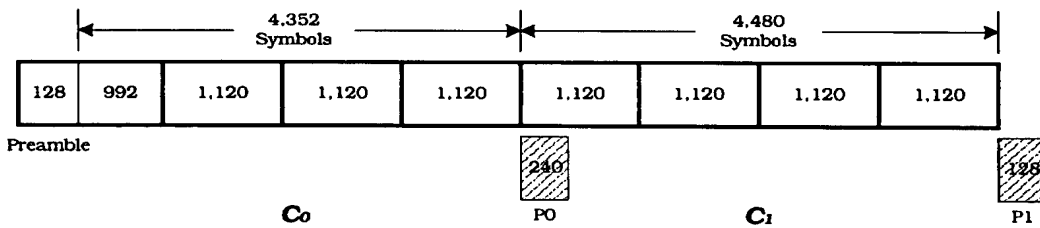
【도 15】



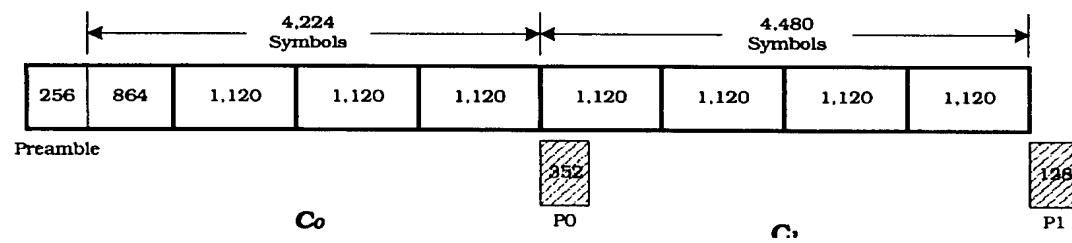
【도 16】



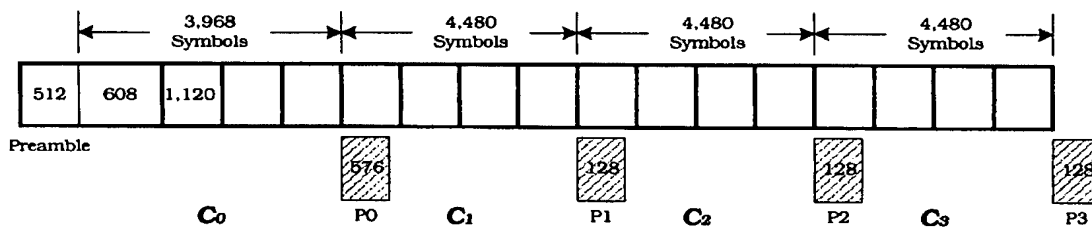
【도 17】



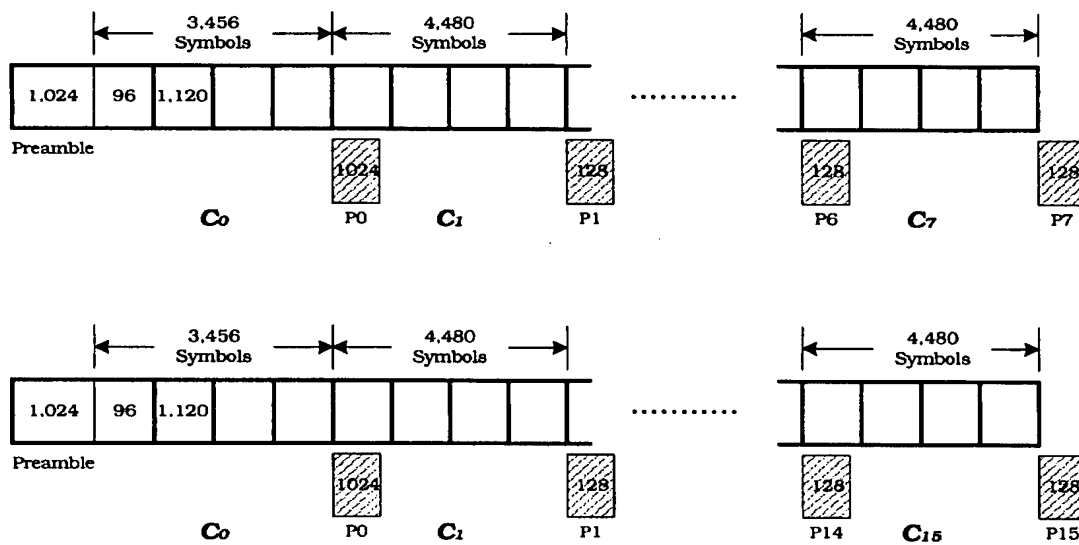
【도 18】



【도 19】



【도 20】



【도 21】

